

FOR WINDOWS AND MAC OS



FileMaker^{PRO 4.0}

DEVELOPER'S GUIDE

© 1998 FileMaker, Inc. All Rights Reserved.

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

<http://www.filemaker.com>

FileMaker and ScriptMaker are trademarks of FileMaker, Inc., registered in the U.S. and other countries. All other trademarks are the property of their respective owners.

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

Mention of third party companies and products is for informational purposes only and constitutes neither an endorsement nor a recommendation.

Contents

Chapter 1: Installing FileMaker Pro Developer Edition

About this <i>Developer's Guide</i>	1-2
Binder application features	1-2
Remove master access	1-2
Specify custom menu items	1-2
Open the runtime application in Kiosk mode	1-2
New APIs	1-3
How FileMaker Pro interacts with plug-ins	1-3
External Function plug-ins	1-4
Naming and registering your plug-in	1-5
External function plug-in examples	1-6
Java class support	1-7
About Java	1-7
How the FileMaker Web Companion and Java classes interact	1-8
About Java classes for FileMaker Pro	1-8
Java resources on the Developer Edition CD-ROM	1-9
Installing the Developer Edition	1-9
What you need	1-9
Installing FileMaker Pro Binder	1-10
Installing additional software	1-10
What end users need	1-11
Using solution files over a network	1-11
Abiding by the license agreement	1-12
Registration and customer support	1-12
What's included on the CD-ROMs	1-13

Chapter 2: Binding solution files

The binding process—an overview	2-1
Before binding	2-1
Binder and runtime terminology	2-2
Specifying the primary file	2-3
Naming the runtime solution	2-4
Assigning the binding key	2-4
Choosing auxiliary files	2-5
Choosing binding options	2-6
Run solution in Kiosk mode	2-7

Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges	2-9
Disable runtime solution closing splash screen	2-10
Use custom “About” script	2-10
Use custom Help script	2-12
Rename Script menu	2-13
Assigning the three-character extension	2-14
Assigning the extension in Windows	2-15
Assigning the extension in the Mac OS	2-16
Changing extensions for cross-platform compatibility	2-17
Specifying a location for solution files	2-17
What happens during binding	2-19
Binding cross-platform solutions	2-20
Changing solutions	2-20

Chapter 3: Preparing and managing runtime solutions

Preparing a solution—an overview	3-1
Considerations before binding	3-2
Documenting your files	3-4
Protecting your solution files	3-5
Policy on decoding and repairing passwords	3-6
Developer responsibilities	3-6
Testing your solution	3-7
Design tips for cross-platform solutions	3-8
Fonts	3-8
Font styles	3-9
Character sets	3-9
Color palettes	3-10
Graphics	3-11
Layout tips	3-11
Moving files between platforms	3-12
Showing the status bar in Windows 95	3-13
Printing tips	3-14
QuickTime movies	3-14
Creating cross-platform scripts	3-15
About the Status (CurrentPlatform) function	3-15
Converting and upgrading solution files	3-16
Converting solution files	3-16
Importing to a new solution file	3-16

Chapter 4: Distributing your bundled solution

Organizing your solution files	4-1
Windows runtime solution files	4-2
Mac OS runtime solution files	4-3
Installing items from the Runtime Files folder	4-3
Contents of the Runtime Files folder	4-4
Choosing the distribution method	4-7
Creating a custom installation	4-7
Compressing your solution	4-8
Documenting your installation	4-8
Installing Win32s	4-9
Creating Win32s diskettes	4-9
Opening your runtime solution	4-10
Updating a solution	4-11
Updating the primary file	4-11
Adding new or changed auxiliary files	4-11
Recovering damaged files	4-12

Appendix A: Feature comparison—runtime application and FileMaker Pro

Menu command comparison	A-1
Runtime application-specific features	A-4
Preferences differences	A-4
Paste As... differences	A-4
Additional limitations	A-5

Appendix B: About the TechInfo database

Navigating the TechInfo database	B-1
Searching the TechInfo database	B-1
Sample TechInfo article	B-3

Index	I-1
--------------	------------

Chapter 1: Installing FileMaker Pro Developer Edition

Welcome to the FileMaker® Pro 4.0 Developer Edition from FileMaker, Inc. The Developer Edition includes software that's designed to help you quickly create and deploy standalone FileMaker Pro runtime solutions for your customers.

Using the FileMaker Pro Binder software, you create a *FileMaker Pro runtime application*—a modified version of the FileMaker Pro 4.0 application. The runtime application binds to your solution files, which you distribute to your end users.

The Developer Edition also includes information about two new application programming interfaces, or APIs, that can increase the power and flexibility of your FileMaker Pro solutions.

- The new External Function plug-in API enables you to write new FileMaker Pro calculation functions using C or C++.
- The new Java API helps you create a Java front-end that communicates with FileMaker Pro.

The Developer Edition also provides database development tools and design tips to boost the effectiveness of your solutions.

The Developer Tools CD-ROM contains:

- databases containing sample artwork and user interface elements
- databases that offer structural design tips
- samples of bound runtime solutions
- third-party database templates and design tips
- FileMaker Pro-related publications and URLs

In addition, the Developer Edition contains the full versions of the FileMaker Pro 4.0 and Home Page 3.0 Web authoring software.

About this *Developer's Guide*

This guide assumes you're familiar with FileMaker Pro and have created database files. For specific explanations of FileMaker Pro functionality, refer to the FileMaker Pro 4.0 onscreen Help and *User's Guide* included with this Developer Edition.

Note The *Developer's Guide* covers both Microsoft Windows and Mac OS versions of the Developer Edition software.

Binder application features

This section outlines key features in the FileMaker Pro 4.0 Binder application.

Remove master access

During the binding process, you can remove access to Define Fields, Define Relationships, Access Privileges, and the ScriptMaker® feature. This prevents end users from modifying the database structure in all solution files.

Important Enabling this feature *permanently* prevents everyone, including you and FileMaker, Inc. employees, from accessing these areas in all the solution files. Save an unbound backup copy of the solution files if you plan on modifying the database structure. See page 2-9 for more information.

Specify custom menu items

You can customize the name of the Script menu, connect a script to a custom menu item in the Help menu, and connect a script to “About <solution name>” (under the Help menu in Windows and under the Apple menu in the Mac OS). See “Choosing binding options” on page 2-6 for more information.

Open the runtime application in Kiosk mode

Kiosk mode allows the runtime application to perform without showing operating system and application interface elements. Like a presentation or slide show, the menu bar and menu choices disappear and the screen surrounding the primary solution file is black.

Kiosk mode is ideal for:

- running your solution files in a retail store, similar to a bookstore directory or gift registry database
- creating a self-running demonstration or survey
- disabling access to the operating system while the runtime application is in use

See “Run solution in Kiosk mode” on page 2-7 for information and design tips.

New APIs

In addition to the Binder application, the Developer Edition contains new APIs that allow you to create:

- External Function plug-ins
- Java applications and applets that communicate via the FileMaker Web Companion plug-in

Note The FileMaker Web Companion is a plug-in included with FileMaker Pro 4.0 that lets you publish your databases on the Web. The FileMaker Web Companion acts as both a web server, delivering web pages to a web browser, and as a CGI application, handling the interaction between the database and the browser.

The Java classes require the FileMaker Pro 4.0 application. The runtime application does not offer support for the FileMaker Web Companion.

How FileMaker Pro interacts with plug-ins

When the FileMaker Pro application is opened, it loads any plug-ins that are enabled in Preferences and are residing in the proper folders. In Windows, plug-ins should be in the System folder inside the FileMaker Pro 4.0 folder. In the Mac OS, plug-ins should be located in the FileMaker Extensions folder.

External Function plug-ins

FileMaker Pro 4.0 supports a new API that lets you create External Function plug-ins that add functionality to solutions through the FileMaker Pro calculation engine.

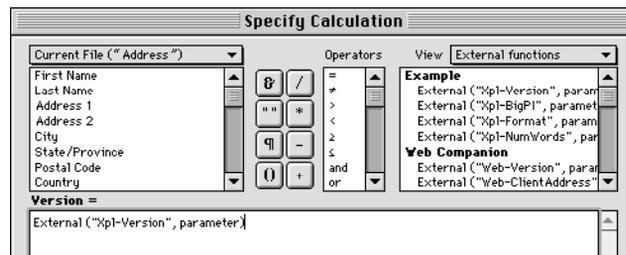
For example, you can create plug-ins that:

- rename, move, or delete files
- open cash register drawers
- maintain activity logs

You can develop plug-ins for Windows or the Mac OS in C or C++ to process data in FileMaker Pro calculations.

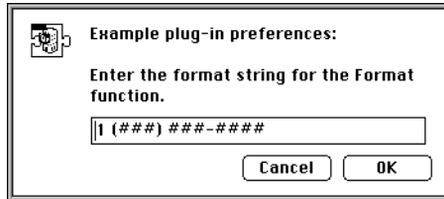
Note In order to write a FileMaker Pro external function plug-in, you'll need to be proficient in a programming language such as C or C++.

Both you and your end users access the plug-in through the External calculation function. The External function uses two parameters: the name of the external function to call and a single parameter for that function to act upon.



External function in calculation dialog

You can program your plug-in to display options in the Plug-Ins panel of the FileMaker Pro Application Preferences dialog box. This lets end users enable or disable specific plug-ins, as well as configure any special features you may have designed.



Preferences dialog box from example plug-in

Important FileMaker, Inc. does not provide technical support for third-party plug-ins. Be sure to test your plug-ins carefully.

For more information on external function plug-ins, refer to EFP Documentation.pdf: Developer Edition CD\FileMaker Pro APIs\External Function Plug-in\EFP Documentation.pdf.

Naming and registering your plug-in

Every external function plug-in name must begin with a unique three- to five-character signature. The Full Example plug-in functions all share the XPL signature, followed by the rest of the function name (for example, Xpl-Version). In order to guarantee the uniqueness of your signature, it's important to register your plug-in with FileMaker, Inc. on the FileMaker, Inc. web site (<http://www.filemaker.com>).

Note FileMaker, Inc. has reserved all external function plug-in names beginning with “Web” or “FM”.

In Windows, each plug-in must use .fmx as the three-character extension.

The Mac OS file type code should always be “FMXT”. The four-character signature acts as the creator code. You should register your creator code with Apple Computer, Inc. to ensure that it is unique. Contact Apple Developer Support or visit their web site (<http://www.apple.com>) for more information.

For information on naming and registering external function plug-ins, refer to the EFP Documentation.pdf document in the External Function Plug-in folder.

External function plug-in examples

There are three sample plug-ins included with the FileMaker Pro Developer Edition:

- a template example, which is a good starting point for plug-in development
- a fully functioning plug-in, which has documentation and commented code
- a folder of external function plug-ins created by third-party developers

The template plug-in includes a function that converts characters to their ASCII equivalents (and vice versa). For more information, see `Read Me-EFP Template: Developer Edition CD\FileMaker Pro APIs\External Function Plug-in\FileMaker, Inc. Examples\Template\Read Me-EFP Template.`

The fully functioning plug-in features four calculation functions:

- The `Xpl-Version` function returns the plug-in version number.
- The `Xpl-BigPI` function returns pi to the fourteenth digit.
- The `Xpl-Format` function is useful for moving numbers into a phone number format. It also provides an example integrating the FileMaker Pro Application Preferences dialog box with a plug-in.
- The `Xpl-NumWords` function returns the text equivalent of a number (similar to a checkbook format). Although this calculation could be created within FileMaker Pro, the result can be calculated more quickly by using the External function plug-in.

For more information, see `Read Me-Full Example: Developer Edition CD\FileMaker Pro APIs\External Function Plug-in\FileMaker, Inc. Examples\Full Example\Read Me-Full Example.`

Java class support

With the new Java API, you can enhance databases you publish on the Web via the FileMaker Web Companion. The Developer Edition also contains information and sample Java applets to help you use the new API.

About Java

Java is an object-oriented programming language for creating and distributing online applets on the Web. With Java, you can create more sophisticated and useful web sites than you can using HTML alone.

Java has some advantages over other programming languages such as C and C++:

- Java code can be written to be platform-independent. After you compile your code into class files, users can view your Java applets on any computer with a Java-enabled web browser. If you develop cross-platform FileMaker Pro solutions, this will save you the frustration of developing twice for platform-specific technologies.
- Because Java is geared for Web usage, it has built-in security features and built-in functionality such as TCP and UDP networking classes. Due to these security features, all FileMaker Pro databases communicating with a Java applet must be using the same IP number.

Using Java, you can design a front-end for your FileMaker Pro 4.0 databases that:

- accesses data from multiple FileMaker Pro databases
- performs any database activities possible with interactive HTML and CDML forms, such as searches and creating, updating, and deleting records

An example of a Java applet used in FileMaker Pro 4.0 is the book icon in the Instant Web Publishing interface. On the Web, users can click the book icon in the status area to move from record to record, just as they do in a stand-alone database. For an example of how Java classes interact with FileMaker Pro, review the examples and documentation in \Developer Edition CD\FileMaker Pro APIs\Java Class Library.

Note Java has a steeper learning curve than HTML. You should be familiar with programming before writing a Java applet.

For general information about Java, visit the Sun Microsystems, Inc. web site at <http://www.javasoft.com>.

How the FileMaker Web Companion and Java classes interact

The FileMaker Web Companion can serve information from FileMaker Pro databases to Java applets. Applet programmers take advantage of this by utilizing the Java Class Library in FileMaker Pro 4.0. The library is a set of Java classes FileMaker Pro supports that are compiled with the programmer's Java applet. These classes process FileMaker Web Companion requests, send them to the computer where FileMaker Pro 4.0 is publishing a database, and pass the results back to the applet.

Java classes communicate with the FileMaker Web Companion using standard HTTP (Hypertext Transfer Protocol). The classes formulate a request string based on the parameters the applet programmer selects. Then the classes send the request to the FileMaker Web Companion. The response is generated and sent back, but instead of returning HTML, a proprietary format is sent back and processed by the `FMProResponse` class. The applet then chooses how to display the data.

About Java classes for FileMaker Pro

There are three main Java classes you can use to communicate with the FileMaker Web Companion:

- `FMProRequest`
- `FMProProxy`
- `FMProResponse`

`FMProRequest` specifies the action the FileMaker Web Companion will take. You specify the name of the database, layout, and fields here.

Use the `FMProProxy` class when a request is ready to be processed. The `FMProProxy` class takes an `FMProRequest` object, sends the request data to the computer where FileMaker Pro is publishing databases, and receives the response from the server. The `FMProProxy` object then converts the response data into an `FMProResponse` object.

Then use the `FMProResponse` class to retrieve information from the result object. The information you have access to includes:

- a result code indicating if the request was successfully processed
- the number of records returned
- the total number of records in the database

- the names, field types, and value lists (if applicable) for each of the fields on the specified layout, as well as field data
- record IDs for the returned records

This information can be retrieved using the various methods belonging to the response object.

Java resources on the Developer Edition CD-ROM

There are two sample Java applets included with the FileMaker Pro Developer Edition:

- FMBanner Java applet, a simple Java applet that displays a textual or picture-based advertisement based on data from a FileMaker Pro database.
- The FMMemoPad applet, which allows you to browse, edit, and create new records in a FileMaker Pro database.

See `default.htm` for a reference of FileMaker Java classes: `Developer Edition CD\FileMaker Pro APIs\Java Class Library\Java Documentation\default.htm`.

Installing the Developer Edition

This section provides instructions for installing the FileMaker Pro Binder application on your system.

What you need

To run FileMaker Pro Binder in Windows 95, Windows 98, or Windows NT 4.0 or later, you need the following minimum equipment and software:

- an Intel-compatible computer with a 486/33 processor
- a hard disk
- 8 MB of RAM (16 MB RAM recommended with Windows NT)
- a CD-ROM drive
- solution files developed in FileMaker Pro 4.0 or 3.x

Note Windows 3.1 and Windows NT 3.51 are not supported by the Binder application. You can develop your solution in Windows 98, Windows 95, or Windows NT 4.0 or later and run it on end users' Windows 3.1 and Windows NT 3.51 systems.

To run FileMaker Pro Binder on a Mac OS-compatible computer, you need the following minimum equipment and software:

- any Macintosh, Power Macintosh, or compatible computer running System 7.1 or later
- a hard disk
- 4 MB of RAM (8 MB RAM recommended)
- a CD-ROM drive
- solution files developed in FileMaker Pro 4.0 or 3.x

Note FileMaker, Inc. recommends using the latest update to the operating system, which is available from Apple Computer, Inc.

To distribute the runtime application on floppy disks, you need to purchase a compression utility or installation program, which stores the runtime application and bound solution files on several disks. Your end users will need the instructions and software to decompress the runtime solution on their hard disk. Trial installation applications are available in the Developer Tools CD-ROM in the Third-Party Tools folder.

Installing FileMaker Pro Binder

To install the FileMaker Pro 4.0 Binder application in Windows or the Mac OS, copy the FileMaker Pro Binder folder to your hard disk. You can also run the FileMaker Pro Binder application directly from the Developer Edition CD-ROM.

Important In Windows, when copying files from the CD-ROMs, you will need to change the file attributes so the files are not read-only format. Right-click on the file you want to modify and choose Properties. Deselect the Read-only checkbox.

Installing additional software

To install FileMaker, Inc. applications such as FileMaker Pro 4.0 or Home Page 3.0 in Windows, double-click the Setup.exe icon. In the Mac OS, double-click the Start Here icon (FileMaker Pro 4.0) or the Claris Home Page Installer icon.

To copy additional files, such as tools to enhance your solution's appearance or sample External Function plug-ins and Java applets, drag the files to your hard disk.

What end users need

To run your solution in Windows 3.1, Windows NT, Windows 95, or Windows 98, your end users need the following minimum equipment:

- an Intel-compatible 486/33 or faster computer with at least 8 MB of RAM
- a hard disk

Note Windows 3.1 requires Win32s 1.30, available in the FileMaker Pro 4.0 folder on the Developer Edition CD-ROM.

Additionally, Windows NT 3.51 and Windows 3.1 require that the 4winsys folder be included in your solution distribution. See “Windows runtime solution files” on page 4-2 for details.

To run your solution in the Mac OS, your end users need the following minimum equipment and software:

- any Macintosh, Power Macintosh, or compatible computer running System 7.1 or later
- a hard disk and 4 MB of RAM (8 MB RAM recommended)

Using solution files over a network

End users cannot access solution files over a network using the runtime application. Unlike the FileMaker Pro 4.0 application, the runtime application does not support peer-to-peer or client/server networking.

To share runtime solution files over a network, each end user must open the solution files using his or her own copy of the FileMaker Pro 4.0 application. For optimal performance, they should host the solution files using FileMaker Pro Server for Windows NT or for the Mac OS.

For information about volume license sales, visit the FileMaker, Inc. web site at <http://www.filemaker.com>.

Warning Not all runtime application features are available in FileMaker Pro 4.0, such as:

- running solutions in Kiosk mode
- specifying custom About, Help, and Script menu items

When a solution file is opened in FileMaker Pro 4.0, the Kiosk mode interface and custom menu items are replaced by the FileMaker Pro 4.0 interface.

Abiding by the license agreement

The FileMaker Pro Developer Edition license agreement allows you unlimited, royalty-free distribution of an unlimited number of runtime solutions. However, there are several terms and conditions in the license agreement you must abide by. Among these terms and conditions are the following:

- You must provide all of the end-user technical support.
- You must distribute a license agreement with your bundled solution. A sample license agreement is included on the Developer Tools CD: `\Developer Tools\FileMaker Examples and Tools\Runtime Solution Example\Sample License File`.
- You must provide an “About” layout that includes your name and the telephone number for your technical support. For more information, see “Use custom ‘About’ script” on page 2-10.

Please review the Developer Edition license agreement, available in the Developer Edition and Developer Tools CD-ROMs, for the complete terms and conditions before distributing your runtime solutions.

Registration and customer support

If a product registration card was included with your software purchase, we encourage you to take the time to complete and mail it to us. In some products an electronic version of the registration card is included for your convenience.

Please refer to our web site at <http://www.filemaker.com> for information concerning customer and technical support. At our web site you will find the FileMaker, Inc. Service Directory (detailing the service options available to U.S. and Canadian customers), answers to frequently asked questions, and access to extensive software libraries used by our technicians to help find solutions. If you do not have access to the Web, you may learn about the service options available to U.S. and Canadian customers by calling 800-965-9090. For customers outside the U.S. or Canada, please refer to the Technical Support brochure included with the software.

What's included on the CD-ROMs

The FileMaker Pro 4.0 Developer Edition product comes with two CD-ROMs:

- the Developer Edition CD-ROM
- the Developer Tools CD-ROM

The following table outlines the general contents of the Developer Edition CD-ROM. For more information on the CD contents, see the Developer Edition CD Contents document, located in the Developer Edition CD-ROM.

Folder name	Description
FileMaker Pro Binder	Contains the FileMaker Pro Binder application
FSA Program	Contains information about the FileMaker Solutions Alliance program
FileMaker Pro APIs	Contains sample Java applets, the Java Class Library reference, sample code, and documentation for the external function plug-in APIs
FileMaker Pro 4.0	Contains full retail versions of the FileMaker Pro 4.0 application for both Windows and the Mac OS
FileMaker Server Trial	Contains trial versions of FileMaker Pro Server 3.0 for Windows NT and the Mac OS
Home Page	Contains full retail versions of the Home Page web authoring application for both Windows and the Mac OS
Product Updaters	Contains product updaters for FileMaker Pro 3.x and FileMaker Pro Server for Windows NT and the Mac OS
Product & Support Information	Contains FileMaker, Inc. product information and Technical Support information, including the TechInfo database

Note To access the Windows versions of software, insert the CD-ROM on a computer with the Windows operating system. To access the Mac OS versions, you must use a computer with the Mac OS operating system.

The Developer Tools CD-ROM contains a number of useful FileMaker Pro 4.0 files that you can use as templates, for artwork, or for database design ideas.

The following table outlines the general contents of the Developer Tools CD-ROM. For more information on the CD contents, see the Developer Tools CD Contents document, located in the Developer Tools CD-ROM.

Folder name	Description
FileMaker Examples and Tools	Contains eight folders (below) including sample artwork, database design examples, and general resources
FileMaker Examples and Tools\Artwork Sampler	Contains over 2,500 pieces of art that can be incorporated into your database as banners, buttons, background art, and general clip art
FileMaker Examples and Tools\FileMaker Tools	Contains sample buttons, layouts, and interface elements that you can use to construct dialog boxes in your FileMaker Pro solutions
FileMaker Examples and Tools\Creating Dynamic Buttons	Shows you how to create multi-state buttons (for example, a “depressed” and a “non-depressed” button)
FileMaker Examples and Tools\Calendar Navigator	Shows you how to incorporate calendars into FileMaker Pro databases
FileMaker Examples and Tools\Runtime Solution Example	Example of a runtime solution, including an “About” layout and a sample Help layout
FileMaker Examples and Tools\Kiosk Solution Example	Example of a FileMaker Pro solution bound in Kiosk mode
FileMaker Examples and Tools\Relational Example	Example of a relational FileMaker Pro database
FileMaker Examples and Tools\URL Register	Contains over 100 useful FileMaker Pro-related web sites
Third-Party Tools	Provides FileMaker Pro design tips, FileMaker Pro templates, trial installation, point-of-sales, and operating system automation software

Chapter 2: Binding solution files

This chapter describes how to create the bundled solution you'll distribute to your end users. It assumes you have developed and thoroughly tested the solution files. See chapter 3, "Preparing and managing runtime solutions," for more information.

The binding process—an overview

To create a runtime solution using the Binder application, you:

1. Specify the solution's primary file.
2. Name the runtime application.
3. Assign a binding key.
4. Add any auxiliary files to the runtime solution.
5. Choose additional binding options, such as Kiosk mode.
6. Assign the three-character extension to solution files.
7. Specify the folder where the runtime application and solution files will be copied.

If your solution will be run in Windows, bind it in Windows. If your solution will be run on the Mac OS, bind it in the Mac OS. For cross-platform solutions (solutions running in both Windows and the Mac OS), bind the files first in one platform, then in the other. For more information, see "Binding cross-platform solutions" on page 2-20 and "Design tips for cross-platform solutions" on page 3-8.

Before binding

FileMaker, Inc. recommends Before you bind, keep the following suggestions in mind:

- Save the original copy of all files before binding. This is especially important if you bind the solution in Kiosk mode or select the Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges option, since both features could render the solution files permanently unmodifiable.
- Close all solution files. To bind a file that is already being hosted on a network, close the file from the host computer before binding.

- Use a binding key you will remember. If you need to add or update auxiliary files later, rebind the file using the same key. When developing a cross-platform solution, use the same key when you bind the solution in Windows and in the Mac OS.

Important For additional considerations before binding, see chapter 3, “Preparing and managing runtime solutions.”

Binder and runtime terminology

This guide uses the following terms when describing the process of creating runtime applications:

Term	Refers to
FileMaker Pro Developer Edition	The Binder application and other files on both Developer Edition CD-ROMs
FileMaker Pro application	The standard version of the FileMaker Pro application
FileMaker Pro Binder	The application that generates the FileMaker Pro runtime application and binds it to your solution files
FileMaker Pro runtime application (or “runtime application”)	The runtime engine you distribute with your solution files
Solution files	Your primary file and any auxiliary files
Primary file	The database file that the runtime application opens first
Auxiliary file	Any additional files, such as import files and relational files
Bundled solution	Your solution files, the runtime application, and additional support files such as DLLs on Windows, optional data and graphics file filters, and spell-checking software
Bind	The process of creating the runtime application and linking it to the solution files
Binding key	A code used when binding the solution files to internally link the solution files with the runtime application
Three-character extension	Associates solution files with the runtime application
Kiosk mode	A unique interface that hides operating system elements and FileMaker Pro standard menus and commands
Cross-platform file	A file that has been designed to run in both Windows and the Mac OS

Specifying the primary file

A runtime solution can consist of only one primary file, or a primary file and a set of auxiliary files. The primary file is the database that the runtime application opens first. While it can contain data, the primary file frequently consists of a “Main Menu” layout containing buttons that navigate end users to other database files (auxiliary files). It should also contain an “About” layout and a sample Help script. Auxiliary files interact with the primary file through relationships or external scripts.

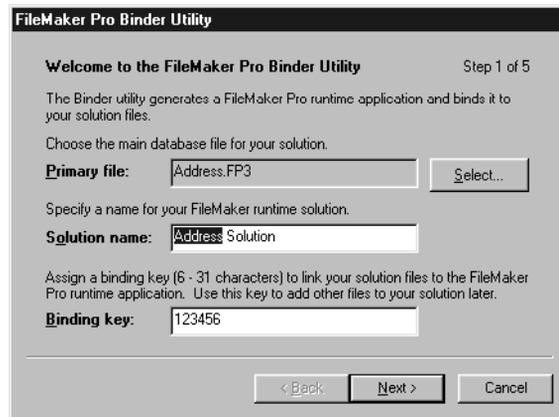


fmbinder.exe

Binder application icon

To specify the primary file for your runtime solution:

1. Double-click the FileMaker Pro Binder application icon in the Developer Edition CD\FileMaker Pro Binder folder.



The first panel in the Binder application

2. Click the Select button to select your primary file.
3. Browse through the contents of your hard disk until you locate the file you want to be the primary file and select it.
4. Click the Open button (Windows) or the Add button (Mac OS).

Tip You can also use “drag and drop” to specify the primary file by dragging the primary file on top of the FileMaker Pro Binder application or into the first panel of the FileMaker Pro Binder application.

Naming the runtime solution

Specify a name for your runtime solution in the Solution Name text box. The solution filename becomes:

- the name of the runtime application
- the name of the solution folder where bound database files are copied

To name the runtime solution:

1. In the first panel of the FileMaker Pro Binder application, click in the Solution Name text box.
2. Type a name for your runtime application.

You do not need to type a filename extension until later in the binding process.

Tip If you specify the primary file first, the primary filename becomes the runtime application name, followed by “Solution”.

When choosing filenames for your solutions, consider the platforms on which your solution will be run so your scripts and lookups will work properly. If your solution will be run on Windows 3.1, use the MS-DOS convention of an eight-character filename.

Platform	Filename support
Mac OS	31 characters
Windows 95	255 characters, including path
Windows NT	255 characters, including path
Windows 3.1	8 characters plus 3-character filename extension

Windows filenames may not start with a space. Additionally, for cross-platform compatibility, do not use the following characters in file names: period (.), quote mark ("), slash (/), backslash (\), brackets([]), colon (:), semicolon (;), vertical bar (|), equal sign (=), or comma (,).

Assigning the binding key

The binding key links the runtime application to the solution files and ensures that the bound runtime application will only open the appropriate solution files.

To specify the binding key:

1. In the first panel of the FileMaker Pro Binder application, click in the Binding key text box.

By default, the binding key is “123456”.

2. Specify the binding key for the runtime application.

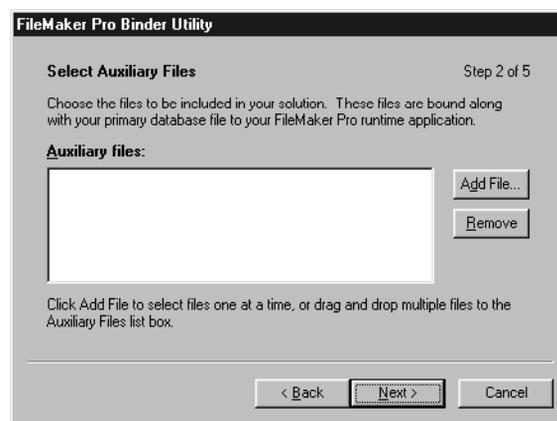
The binding key must be between 6 and 31 characters and is case-sensitive in the Mac OS. When creating solutions for both the Windows and Mac OS platforms, use the same key for both solutions.

3. Click Next to move to the next Binder panel.

Important Use a binding key you’ll remember and record it in a safe place. To add or modify auxiliary files to an existing runtime solution, you’ll need to use the same binding key. If you forget your binding key and would like to change a runtime solution, you’ll need to rebind using a new binding key and redistribute the entire solution, including a new runtime application.

Choosing auxiliary files

Auxiliary files are other FileMaker Pro files that the primary file interacts with (for example, related files). You may bind as many files as you like, but the runtime application, like FileMaker Pro 4.0, can only open up to 50 files at once.



The second panel in the Binder application

To add auxiliary files to the solution:

1. In the second panel of the FileMaker Pro Binder application, click the Add file button.

You can specify all the solution files at once by dragging them on top of the FileMaker Binder application. You can also drag auxiliary files directly into the second Binder application panel.

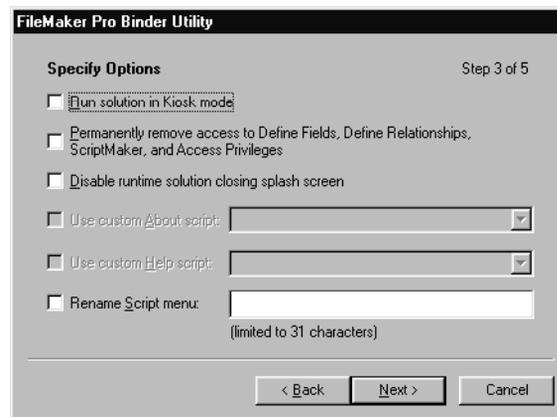
2. Browse through the contents of your hard disk until you locate the files you want to be auxiliary files.
3. Click the Open button (Windows) or the Add button (Mac OS).
4. Click Next to move to the next Binder panel.

Tip To ease navigation for your end users, you can place a “Main Menu” or “Home” button in all auxiliary files so end users can easily return to the primary file. Or, create a startup script that moves the end user to the primary file in each auxiliary file.

Choosing binding options

There are a number of binding options you can set that change the behavior of the runtime application and solution files.

Important Some options, such as selecting the Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges option or running the solution in Kiosk mode, will render parts of the solution files unmodifiable after binding.

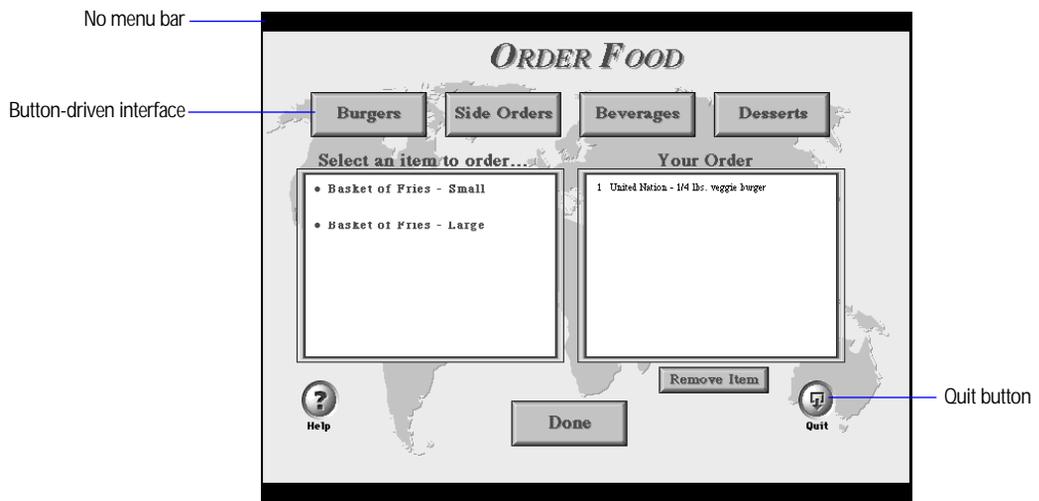


The third panel in the Binder application

Run solution in Kiosk mode

Select Run solution in Kiosk mode if you want to provide a special interface for your solution. As in a presentation application or slide show, many of the standard system interface components are removed, such as the menu bar and menu commands, and the screen area around the solution is black.

To work with a sample bound solution file with Kiosk mode enabled, open the Kiosk Solution Example in the FileMaker Pro Examples and Tools folder on the Developer Tools CD-ROM.



Solution Example in Kiosk mode

Important In order to execute any menu command, such as creating new records, sorting, and closing the solution files, you must create buttons on available layouts to perform the desired menu commands. For sample buttons, see the Artwork Sampler and FileMaker Tools files in the FileMaker Pro Examples and Tools folder on the Developer Tools CD-ROM.

The following table describes how to emulate interface elements that are not shown when you run your solution files in Kiosk mode.

To use these interface elements	Do this
Menu commands	Use buttons for all menu commands (for example, Sort, New Record, and Quit)
Zoom controls	Use the Toggle Window or Set Zoom Level script steps
Status area control	Use the Toggle Status Area button
Mode pop-up menu	Use the Enter Browse Mode, Enter Preview Mode, or Enter Find Mode buttons
Vertical scroll bar	Scroll if the layout is longer than one screen or if browsing a View as List layout
Horizontal scroll bar	Scroll if the layout objects are wider than one screen
Window size and position	Set before binding or use a startup script to toggle the window size

Important Kiosk mode is not supported in FileMaker Pro 4.0. When a solution file is opened in FileMaker Pro 4.0, the Kiosk mode interface is replaced by the FileMaker Pro 4.0 interface.

Closing files in Kiosk mode

The only way to close a solution in Kiosk mode is by a script attached to a button that quits the runtime application. If there is no Quit or Exit button available, you can exit the runtime application by pressing Alt-F4 (Windows). In the Mac OS, you must force the runtime application to quit by pressing Command-Option-Esc. This is not recommended because it can cause data corruption or damage open files.

Tip To ensure that end users can always access the primary file, you can:

- provide a startup script in each auxiliary file that opens the primary file
- place a “Main Menu” button in each auxiliary file that returns end users to the primary file

When a solution is running in Kiosk mode in the Mac OS, access to the operating system is restricted. To switch to another application or the operating system, you must exit the runtime application. In Windows, pressing Alt-Tab will move you to another application.

Design tips for Kiosk mode

Start by planning your database design on paper. Decide what will happen when each button is selected, and give users a way to get back to the beginning from each layout.

- If your solution will be run with a touch screen, use large buttons and allow space between buttons.
- Try to limit the number of options available on one screen.
- Preview mode disables buttons; make sure you create a script to return you to Browse mode by choosing Continue or pressing the Enter key. If you have a script that enters Preview mode and end users won't have access to a keyboard, follow it with the Pause/Resume Script step and specify an amount of time the script should remain in Preview mode. Then have the script return to Browse mode.

In Kiosk mode, it is often desirable to center solution files on the screen. To center your files, create a startup script that zooms the window. Using the Toggle Window script step with the zoom option will perform two actions in Kiosk mode:

- snap solution file windows to fit layout objects at the right and bottom edges of the layout
- center the windows on the screen if the window is smaller than the available screen area

Important Before using the Toggle Window script step, perform any script steps that affect the window display area (such as Go to Layout or Toggle Status Area). Once the window area is determined, add the Toggle Window script step.

The Toggle Window script step may cover up a window that has an Exit/Quit button. Be sure that you can navigate to somewhere that allows you to close the solution elegantly.

Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges

In the third panel of the FileMaker Pro Binder application, when you select the Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges option, access to these areas is permanently unavailable in solution files. This prevents end users from modifying the database structure in all solution files.

This option makes solution files behave as if the Access the entire file and Edit scripts checkboxes in the FileMaker Pro 4.0 Define Passwords dialog box were de-selected. The Access the entire file option restricts access to Define Fields, Define Relationships, and changing Access Privileges passwords and groups. Additionally, the textual version of any calculations you've defined (for example, calculation field formulas) is also removed.

Important Selecting this option *permanently* removes database structure in all included solution files, even when they're opened in the FileMaker Pro 4.0 application. This will help protect your file from password hackers. Structural elements of the files cannot be modified by anyone, including FileMaker, Inc. employees. The only way to modify field definitions, relationships, scripts, or access privileges is by returning to a backup copy of the file saved before it was bound.

Consider your end users' long term needs when defining access privileges. Communicate their access levels to them clearly in the "About" layout and follow the rules specified by FileMaker, Inc. See "Policy on decoding and repairing passwords" on page 3-6 and the next section, "Use custom 'About' script," for more information.

Disable runtime solution closing splash screen

Selecting this option prevents the "FileMaker Pro 4.0 Runtime" dialog box from displaying when end users exit or quit the runtime application.

Use custom "About" script

As the Developer Edition license specifies, you must create an "About" layout on all your solution files that provides information for your end users on how to contact you. This helps protect your solutions in the event that your end users approach FileMaker, Inc. for runtime solution passwords. FileMaker, Inc. uses the "About" layout to distinguish solutions that may be owned by developers rather than end users of the retail version of FileMaker Pro. For more information, see "Policy on decoding and repairing passwords" on page 3-6.

To replace the "About" menu item in the runtime application:

1. In the third panel of the FileMaker Pro Binder application, select the Use custom About script checkbox.
2. Select the script name that displays your "About" FileMaker Pro layout.

This replaces the “About FileMaker Pro runtime” menu item in the Help menu (Windows) or in the Apple menu (Mac OS) with the name of the script you specified in the primary solution file.

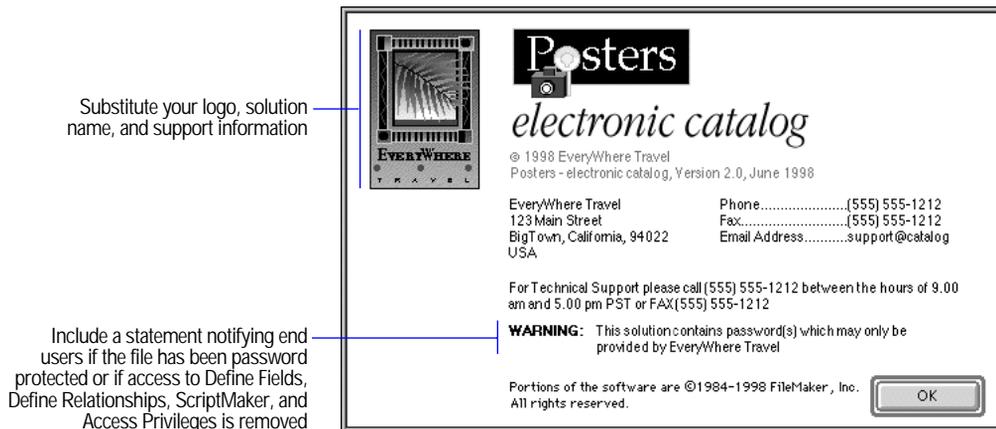
You can also display the “About” layout by:

- Writing a script that displays the layout each time the end user starts the bundled solution.
- Displaying the “About” layout when the runtime solution is opened. To do this, in the FileMaker Pro 4.0 Document Preferences, set the Switch to layout option for the primary file. Select the “About” layout as the layout that appears first.

Important Title the layout “About <solution name>.” Include your logo, name, address, and the telephone number you are using for technical support. You must also include additional information in your “About” layout. See the license agreement for details.

To create an “About” layout:

1. Start with the “About” layout in the Runtime Solution Example in the FileMaker Pro Examples and Tools folder on the Developer Tools CD-ROM.



Sample “About” layout

2. Include your logo, any other graphics, and your company information.

3. Notify end users if you've password protected the solution files or if you've selected the Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges binding option. (See "Policy on decoding and repairing passwords" on page 3-6 for the exact legal wording.)
4. Include a button that lets your end users return to the original layout.

Use custom Help script

Select Use custom Help script to add a Help menu item ("`<solution name> Help`") to the top of the Help menu in Windows and at the bottom of the Help menu in the Mac OS.

To specify a custom Help script:

1. In the third panel of the FileMaker Pro Binder application, select the Use custom Help script checkbox.
2. Select the script name that displays your FileMaker Pro help layout in the primary solution file.

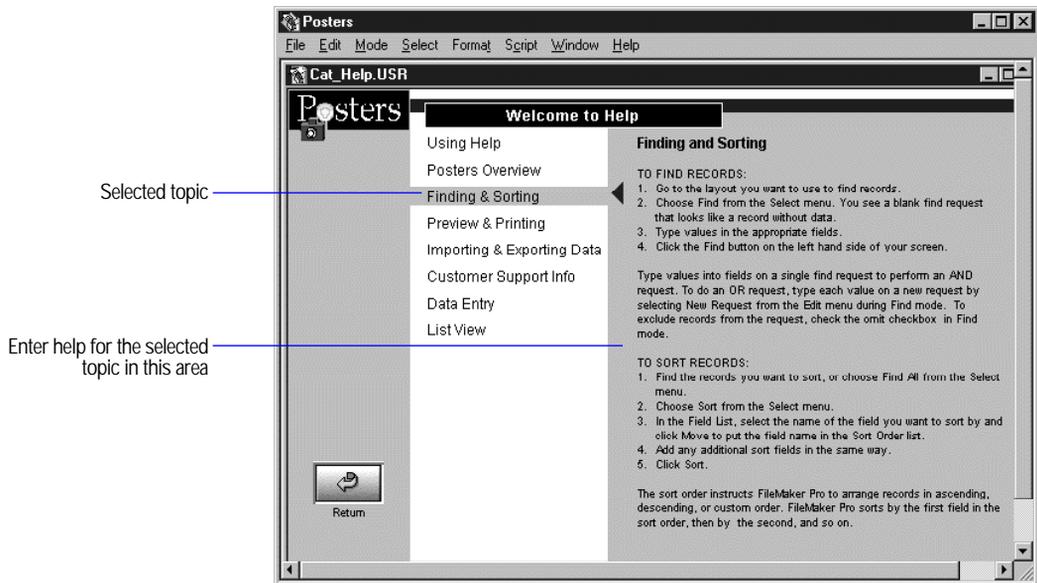
When end users select "`<solution name> Help`" in the runtime application, the specified script executes and displays your Help layout.

Note The FileMaker Pro Help system is not available in the FileMaker Pro runtime application.

To create a help layout:

1. Using FileMaker Pro 4.0, open the `Cat_help.usr` file: `Developer Tools CD-ROM\FileMaker Pro Examples and Tools\Runtime Solution Example`.

The Runtime Solution Example demonstrates a button-driven help system that is integrated into the solution. It provides help on a variety of FileMaker Pro tasks, such as finding, sorting, printing, importing, and exporting. Using FileMaker Pro buttons and commands, users are directed to a separate file where they select help topics from a menu. A button is available to return the user to the main solution file.



2. Edit the layout so that your logo and solution name appear.
3. Select a help topic on the left and then enter the help information on that topic on the right, in the corresponding layout.
4. Make a new layout for each help topic.
5. Modify the Return button script so that it returns end users to your solution.

FileMaker, Inc. recommends Although much of your documentation will be in the Help file, you should create and distribute printed documentation for your solution, in the event that end users are not able to open your solution files. For more information, see “Documenting your files” on page 3-4.

Rename Script menu

Select Rename Script menu to give the Script menu in the runtime application a custom name.

To rename the Script menu in the runtime application:

1. In the third panel of the FileMaker Pro Binder application, select the Rename Script menu checkbox.

2. Type the menu name that replaces the “Script” menu.

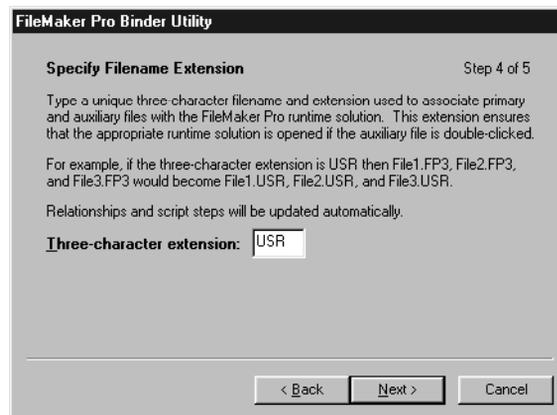
For example, type “<solution name>Reports.” The menu name you specify cannot exceed 31 characters.

3. In Windows, specify a keyboard accelerator by typing an ampersand (&) in front of the character you want to become the accelerator key (for example, Address Re&ports will display as Address Reports).

Once you’ve specified options on the third panel of the Binder application, click Next.

Assigning the three-character extension

The three-character extension associates the solution files with the FileMaker Pro runtime application. This will help your customers identify your solution files and, together with the binding key, ensure that the correct runtime application opens when a solution file is double-clicked.



Fourth panel of Binder application (Windows)

To specify a three-character extension:

1. In the three-character extension text box, assign the extension for your solution or use the default “.usr” extension.
2. Type a unique three-character extension.

If the three-character extension is not unique, it might cause registry (Windows) or desktop (Mac OS) conflicts. For example, if you use the “.fp3” extension and your end users have FileMaker Pro installed on their hard disks, their FileMaker Pro document icons (and application icon in the Mac OS) will change to the runtime icons. Additionally, FileMaker Pro documents will no longer automatically open the FileMaker Pro application.

If you’ve assigned duplicate extensions, you can return the document icons to the FileMaker Pro icon in Windows by discarding the runtime application, opening the FileMaker Pro application, then restarting the computer. In the Mac OS, discard the runtime application, and then rebuild the desktop file.

3. Click Next to move to the next panel in the Binder application.

Important Even with different extensions, solution files can still be opened in the FileMaker Pro application. To prevent end users from modifying solution files, assign passwords using FileMaker Pro 4.0 Access Privileges.

Tip If your end users want to run your solution files with the FileMaker Pro 4.0 application, for example, to share solution files over a network, you can use the Binder application to change their three-character extension. For example, you can rebind a runtime solution that uses the “.usr” extension and change the three-character extension to “.fp3”. The Binder application preserves the internal file specifications used in relationships and scripts that reference other files.

Note The Binder application can create or modify existing three-character extensions, but cannot be used to remove an extension from a solution file.

Assigning the extension in Windows

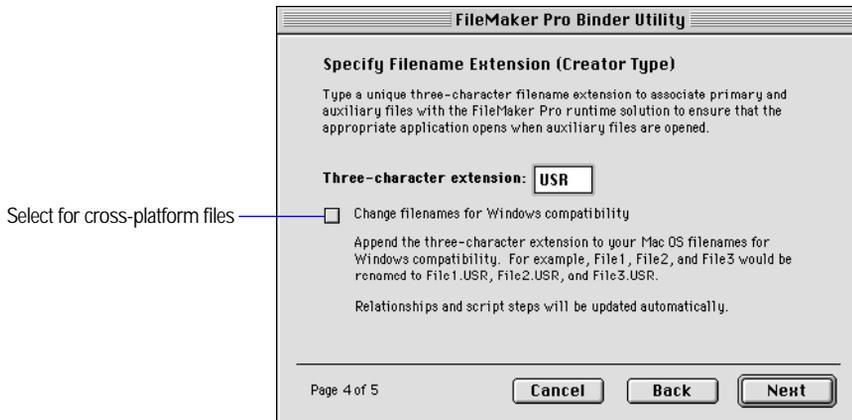
The three-character extension registers your runtime application with the operating system. The extension is used by Windows to determine which application starts when you double-click the primary file. The runtime application adds the extension to all solution filenames during the binding process.

Important You must specify a three-character extension. Choose an extension that is unique to your end user’s computer system.

Assigning the extension in the Mac OS

In the Mac OS, the three-character extension becomes the creator code for the runtime application. The creator code must be unique to ensure that the Finder in Mac OS can determine which application created each document. The creator code is stored in both the solution files and the runtime application.

Because creator codes are four characters, FileMaker Pro Binder inserts an uppercase F after the first character. For example, the default three-character extension “USR” becomes the “UFSR” creator code. Creator codes are case-sensitive.



Fourth panel of Binder application (Mac OS)

Important Creator codes should be registered with Apple Computer to verify that the creator code you choose is unique. If the creator code is not unique, solution files might not open with the appropriate runtime application. You may use the default “USR” three-character extension, as FileMaker, Inc. has registered the “UFSR” creator code with Apple Computer, Inc. Contact Apple Developer Support or visit their web site (<http://www.apple.com>) to register any other creator codes.

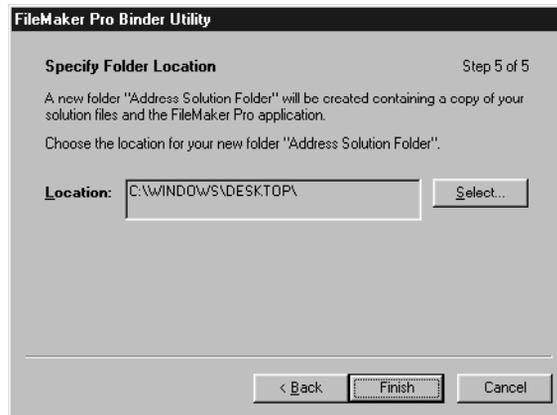
Changing extensions for cross-platform compatibility

If you're binding solution files on the Mac OS and creating a solution to run in both Windows and the Mac OS, select the Change filenames for Windows compatibility option in the fourth binding panel. The Binder application automatically updates all filenames to use the three-character extension specified during binding. Internal file references used in relationships, scripts, and external value lists are changed to interact with the new filenames.

For example, if you bind three files named A.fp3, B.fp3, and C.fp3 and specify a three-character extension of .usr, the files are renamed to A.usr, B.usr, C.usr.

Specifying a location for solution files

On the last panel of the Binder application, you specify the folder where the bound solution files and runtime application will be copied.



Fifth panel of Binder application

To specify the solution folder location:

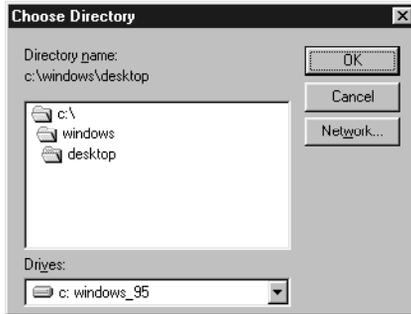
1. In the fifth panel of the FileMaker Pro Binder application, click the Select button.

A dialog box appears.

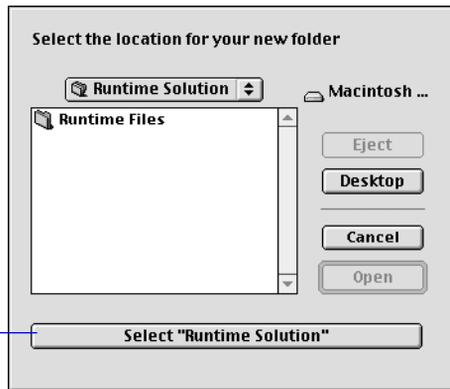
2. Browse through the contents of your hard disk until you locate the folder you want the solution folder to be in.

In the Mac OS, click the folder name to put it in the Select "<folder>" button at the bottom of the dialog box.

3. In Windows, double-click the folder name and click OK. In the Mac OS, click the Select "<folder>" button.



Windows



Mac OS

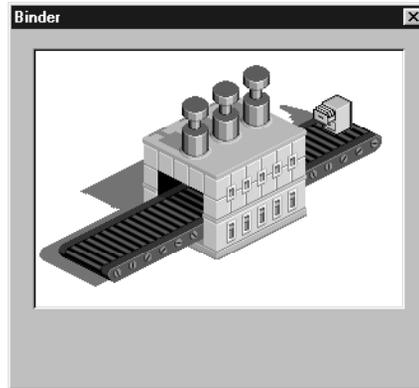
4. In the fifth panel, click the Finish (Windows) or Bind (Mac OS) button to bind the new runtime application to the solution files. Or, click the Back button to review your binding options.

What happens during binding



When you click the Finish button (Windows) or the Bind button (Mac OS), FileMaker Pro Binder performs the following activities:

- saves a copy of the solution files into the folder specified in the fifth panel
- updates the solution files with binding options, such as adding three-character extensions or renaming menu items
- generates a runtime application
- binds the runtime application with the specified solution files



Binder application creating runtime solution

When the binding process has been successfully completed, you see a dialog box updating you on the status of your runtime solution.



Binder application confirmation screen

Binding cross-platform solutions

If you're creating a solution to run on both Windows and the Mac OS, bind the solution files twice: first using FileMaker Pro Binder for Windows, and then using FileMaker Pro Binder for the Mac OS. Use the same binding key on both platforms. Remember that binding keys are case-sensitive in the Mac OS.

When binding solutions in the Mac OS first, select the Change filenames for Windows compatibility option in the fourth panel to automatically update all filenames (and all internal references to filenames) to use the three-character extension specified during binding. This saves you from renaming and respecifying files used in relationships and external scripts. See “Design tips for cross-platform solutions” on page 3-8 for additional cross-platform information.

Changing solutions

To add solution files or change binding options, rebind the files in FileMaker Pro Binder using the same binding key as the distributed solution.

You might also need to rebind your solution if you change the name of the primary file or if you forget the binding key. In these cases, you need to rebind your primary file with the Binder application and distribute the new solution files and runtime application. See “Updating a solution” on page 4-11 for more information.

Chapter 3: *Preparing and managing runtime solutions*

This chapter describes:

- preparing your files before binding
- protecting your files with passwords
- FileMaker, Inc. legal requirements for developers
- testing tips for your solution
- tips for designing solutions that run on Windows and Mac OS computers
- instructions for upgrading runtime solutions

Preparing a solution—an overview

In preparing your solution files, you should implement the following:

- an “About” layout to introduce your solution
- a customized Help system
- scripts/buttons to open and close auxiliary files
- passwords to protect your solution

Here are some general steps to follow in preparing your solution files:

1. Be sure to keep backups of all files before binding.

Although FileMaker Pro Binder copies your files, it is important for you to save backups of the unbound solution files, especially if you are using the runtime application in Kiosk mode or have selected Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges while binding. These features will render the solution files unmodifiable. See “Choosing binding options” on page 2-6 for information on these features.

2. In FileMaker Pro 4.0, design and test the files that make up your solution.

If you’re converting solution files, see “Converting and upgrading solution files” on page 3-16.

3. Add an “About” layout.

You should name your layout “About <solution name>.” Include your name, address, and technical support phone number on the layout—you are responsible for providing technical support for your bundled solution. See “Use custom ‘About’ script” on page 2-10 for more information.

4. Create a Help system that provides usage and navigational tips for your solution.

The Help files contained in the FileMaker Pro application cannot be distributed with the FileMaker Pro runtime application. Explore the sample integrated help system: `Developer Tools\FileMaker Examples and Tools\Runtime Solution Example`. See “Use custom Help script” on page 2-12 for more information.

5. Use the FileMaker Pro Binder application to bind the primary and auxiliary files. See chapter 2, “Binding solution files” for more information.
6. Test your solution for errors.

Go through your solution trying every option, feature, and button. See “Testing your solution” on page 3-7 for a list of potential problems when testing solution files.

Important If your solution will be run in both Windows and the Mac OS, test your solution on both platforms. See “Design tips for cross-platform solutions” on page 3-8 for cross-platform database design advice.

7. Create a custom installer.

Since the runtime application doesn't have an automatic installer, create an installation procedure for your solution files. If you're distributing the bundled solution on floppy disks, you will also need a utility to decompress and install the files on your end user's hard disk. The Developer Tools CD-ROM includes trial installer applications.

Considerations before binding

Here is a list of things to consider and decide as you develop your solution. It's best to address these issues before binding the solution.

- Do you want end users to open and close auxiliary files?

In the FileMaker Pro runtime application, several menu options have been eliminated, including Open and Close. If you want end users to open and close auxiliary files, you must provide scripts in your solution to perform these operations.

The FileMaker Tools and Artwork Sampler database files in the FileMaker Examples and Tools folder on the Developer Tools CD-ROM contain sample buttons you can use to make opening and closing files easier for your end users.

- Is the runtime application operating in Kiosk mode?

If your solution is running in Kiosk mode, the entire interface must be accessible via buttons on the layouts. Include an Exit or Quit button if the file needs to be closed.

- Does the runtime solution contain auxiliary files?

Place a “Main Menu” button or startup script in each auxiliary file that moves end users to the primary file. See “Choosing auxiliary files” on page 2-5 for more information.

- Should you enable Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges?

This option changes the structure of the solution files, making Define Fields, Define Relationships, ScriptMaker, and Access Privileges unmodifiable by FileMaker Pro. No one will be able to access these features, including FileMaker, Inc. employees. See “Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges” on page 2-9 for more information.

- Do you want your end users to be able to modify the database?

You can password-protect solution files before binding to prevent end users from modifying database files. This helps protect the file in case an end user opens your solution with the FileMaker Pro application. Without password protection, end users can print or modify your scripts and field definitions, and make modifications to your solution. See “Protecting your solution files” on page 3-5 for more information.

- Is this a cross-platform solution?

Review “Design tips for cross-platform solutions” on page 3-8 for advice for handling fonts, graphics, and general cross-platform issues. For binding tips, see “Binding cross-platform solutions” on page 2-20.

- How will you provide updates for your end users?

Plan ahead for the time you may want to update your solution files. You can make the process easier for end users by providing scripts in your primary file to export and import their data. See “Importing to a new solution file” on page 3-16 for an overview.

- Which printers will your end users have?

It’s a good idea to set document margins in FileMaker Pro 4.0 if your solution will be printed from a variety of printers.

Documenting your files

Although much of your documentation might be in a custom Help file, you should create and distribute printed documentation for your solution to cover items not appropriate or useful in help.

Here are some suggestions for items to include in the printed documentation:

- how to install your bundled solution
- how to update to new solution files
- how to use your onscreen help
- how to start the solution (see “Opening your runtime solution” on page 4-10)
- what to do in case of a damaged file (see “Recovering damaged files” on page 4-12)
- how to reach you for technical support

You might also want to include the following recommendations:

- Tell your end users not to rename any solution files (except the runtime application), or they may be unable to run your solution.
- Recommend that your end users back up their data regularly. You might want to automate some of the process by including scripts that save copies of the solution files. For more information on automating a backup procedure, search the TechInfo database, included in the Product & Support Information folder on the Developer Edition CD-ROM. For information on using the TechInfo database, see appendix B, “About the TechInfo database.”

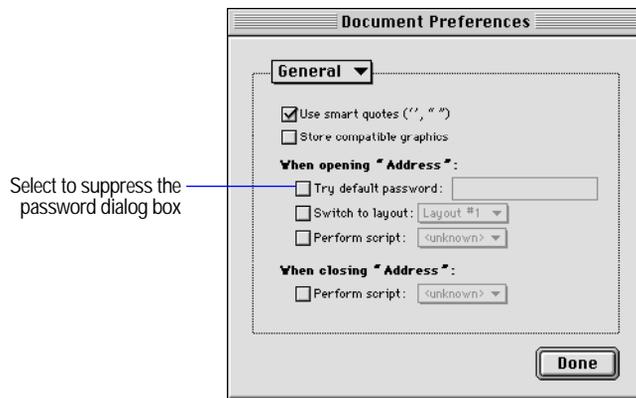
Important In the event that your solution files become damaged, make sure end users have access to your technical support phone number in your printed documentation or in a text file. If the file is damaged, they may not be able to access the “About” layout in your solution to find your phone number.

Protecting your solution files

To prevent end users from modifying the database structure, you should define passwords in the FileMaker Pro files before binding. Solution files can be opened in FileMaker Pro 4.0; if they don't have passwords, end users can access and modify scripts, field definitions, access privileges, and layouts.

You may want to set a master password that gives you access to the entire file and another password that gives end users limited access. To permanently limit access to the structure of the solution files, select the Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges option in the Binder application. See page 2-9 for more information.

Tip If most of your end users will be using the same password to access your solution, you can suppress the password dialog box by selecting the Try default password option in the FileMaker Pro 4.0 Document Preferences.



FileMaker Pro 4.0 Document Preferences dialog box

To open a file using a password other than the default password, hold down the Shift key in Windows or the Option key in the Mac OS while double-clicking the FileMaker Pro runtime application icon.

Policy on decoding and repairing passwords

FileMaker, Inc. has established procedures, and if a customer complies with these procedures, then FileMaker, Inc. will decode passwords, and/or supply data or a repaired file for such customer.

This policy is outlined in the FileMaker, Inc. document entitled “Unlocking Password Protected Files.” This document can be accessed via the support area of our web site: <http://www.filemaker.com> or through the FileMaker, Inc. fax back system: (800) 800-8954 or (408) 987-3900.

FileMaker, Inc. reserves the right to change this policy without notice. If you have specific questions about these procedures, please contact FileMaker, Inc.

Developer responsibilities

If you distribute FileMaker Pro files with passwords or if you have removed master level access, and you do not want FileMaker, Inc. to decode passwords, repair access privileges, and/or return data to a customer who requests such services, you must:

1. Notify your customers in writing and keep a record of such notice that your solutions contain passwords and/or data that can only be provided by you.
2. Every file in your solution must contain an “About” screen accessible from any layout in the database. See “Use custom ‘About’ script” on page 2-10 for more information.
3. The layout name must begin with the word “About”.
4. The “About” screen must contain all of the following items:
 - Solution Name
 - Your company information: company name, address, city, state, postal code, country, and phone number.
 - Your support policy (for example, how and when you are available for technical support).
 - This exact warning “USER WARNING: This solution contains password(s) which can only be provided by the Developer identified above.”
 - If Permanently remove access to Define Fields, Define Relationships, ScriptMaker, and Access Privileges has been selected in the binding options, this exact warning must be included: “USER WARNING: This file is not customizable. Contact the above named developer for information on customizing this solution.”

FileMaker, Inc. is unable to, and will not, investigate any requests to verify the truth of statements submitted by the requester beyond what is required by our procedures. This process has been invaluable to numerous customers who have legitimately been unable to access their valuable files.

The password protection in a FileMaker file should not be viewed as an absolute barrier which will prevent a customer from accessing files. FileMaker, Inc. cannot guarantee that a customer will not be able to identify or bypass the password without any assistance from FileMaker, Inc. We therefore recommend that you take appropriate steps to protect your consulting and development efforts without relying solely upon the password.

If you have a dispute with your customer, you must resolve this dispute directly with the customer. FileMaker, Inc. is unable to, and will not, attempt to resolve such disputes.

Testing your solution

You should verify the functionality of your solution by testing it thoroughly before and after binding it to the FileMaker Pro runtime application.

Here are some suggestions for ensuring the quality of your solution:

- Verify every function and option in your solution. If you're developing a cross-platform solution, test it in both Windows and the Mac OS platforms. See the next section, "Design tips for cross-platform solutions," for more information.
- Make sure your solution does not use a standard feature that is hidden or disabled in the runtime application. See appendix A, "Feature comparison—runtime application and FileMaker Pro" for more information.
- Verify that the scripts and buttons work as expected. This is especially important if you are running your solution in Kiosk mode.
- Verify your installation procedures and test your documentation.
- Verify that your solution layouts display well on monitors with different color capabilities and resolutions.
- Verify that your solution layouts display well on the smallest size monitor your end users may be using.

- Test your solution with actual data. This is especially important if end users are upgrading from earlier versions of the runtime application and need to import data into new solution files.
- Make sure all the auxiliary files and DLLs (Windows) are present.
- Show your solution to intended end users to uncover any usability issues.
- Install your bundled solution to a completely different computer to verify that all the files associated with the primary file can be found.
- If you're removing master access or assigning passwords, test all access levels. Make sure your end users are aware of the level of access you're providing. See "Use custom 'About' script" on page 2-10 and "Policy on decoding and repairing passwords" on page 3-6 for more information.

Important You should keep an unbound version of any solution files, especially if you've removed master access.

Design tips for cross-platform solutions

If you're developing a solution to be run on both IBM-compatible and Mac OS-compatible computers, there are many issues that require planning and consideration (for example, font technology, colors, and design conventions). For information on creating cross-platform runtime solutions, see "Binding cross-platform solutions" on page 2-20.

Fonts

To ensure a consistent look for files across platforms, try to use the same font technology, for example TrueType. It is best to use fonts with identical names, styles, and metrics on both platforms. When fonts used to view a file are different from the ones used to create the file, there can be problems with word wrap and placement.

Most font vendors supply Windows and Mac OS versions of the same font. Contact your font vendor for more information.

When you move a file from one platform to another, if the font used to create the file isn't available, another font will be substituted. In FileMaker Pro 4.0 for Windows, you can specify how fonts are substituted in the Microsoft Windows Registry Editor.

You can add more fonts to the Registry Editor to include more font substitutions. If there is no match and no substitute is defined, FileMaker Pro changes the font to Arial.

In the Mac OS, FileMaker Pro attempts to substitute a similar font.

This Windows font	Is substituted with this Mac OS font
MS Serif	Times
Times New Roman	Times
Times	New York
Tms Rmn	Times
Courier New	Courier
Courier	Monaco

Font styles

The Windows platform doesn't offer outline and shadow font styles. If you want a shadowed effect for titles or headings, simulate the look by positioning two copies of a text block so that the foremost copy appears to cast a shadow against your layout:

1. Add text to your layout.
2. Set its text color to a shade of gray.
3. Turn on the grid if it isn't already active.
4. Duplicate the text object.
5. Set the color to black.
6. Move the duplicate back over the original.
7. Press the up-arrow key once, then the right-arrow key once.

You can vary the offset to achieve whatever shadow effect you prefer.

8. Check on both platforms to be sure the shadow effect appears as you intended.

Character sets

When sharing files between the Windows and Mac OS platforms, use characters that exist in both character sets when available. While international characters are usually available, many of the "high ASCII" characters don't match cross-platform. Keep in mind that special characters are handled differently across fonts.

The following are examples of characters that won't display properly across platforms:

Windows-only characters:

- 1/4 symbol
- 1/2 symbol
- 3/4 symbol

Mac OS-only characters:

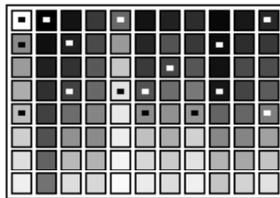
- ≠ (not equal sign)
- ≥ (greater than or equal to)
- ≤ (less than or equal to)
- √ (square root/check mark)

Color palettes

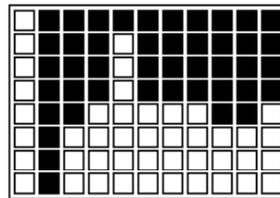
When 256 or more colors are available, FileMaker Pro will offer an 88-color palette that is virtually identical across platforms. (Close substitutes are used for the 13 colors that do not match exactly.)

Windows computers using a standard VGA driver will only display 16 colors. Some older Mac OS computers may also be limited to 16 colors. The 16-color palette in FileMaker Pro varies slightly between Windows and the Mac OS, depending on the Windows color scheme you are using.

If you are building your files on a 256-color computer, you might want to use colors that will map well to 16-color systems. It also helps to know which colors map to black and which map to white for monochrome displays.



Dotted colors are available on 16-color systems



Palette indicating which colors map to black or white

Graphics

If you're storing and displaying graphics across platforms, be sure to select the Document Preference for Store compatible graphics before importing the graphic. Then two copies of the graphic image will be stored: the original version (for example, bitmap, metafile, or GIF) as well as a PICT file format.

Graphics containing gradients

Graphics with gradients that are imported and stored as bitmaps will redraw faster onscreen than graphics imported and stored as PICT images. Additionally, PICT images containing gradients may have some quality degradation when displayed on Windows.

Note Pasting a graphic into a FileMaker Pro container field will store the graphic as a PICT image. To store a graphic as a different file format, use the Import Picture command in the File menu.

If your graphic image has an undesirable white border surrounding an irregularly shaped graphic, you should create a mask for the bitmap image. Refer to the documentation that came with your graphics program for more information.

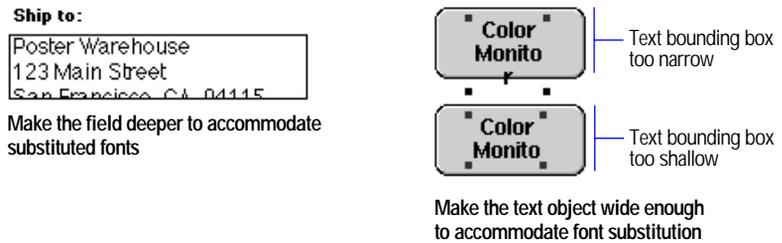
Layout tips

Keep these design tips in mind when designing a layout that will display in both Windows and the Mac OS.

Leave plenty of room

Even when FileMaker Pro is able to match fonts, there can be subtle differences in character width and line spacing when fonts are substituted on another platform. FileMaker Pro supports fixed text object sizes, so that you can resize a text object to be longer than the text within it. This allows you to anticipate changes in font size. Resize your text objects so they are a little wider than the default to prevent text from wrapping to a second line when you take a file cross-platform.

Note Editing a text object will cause the text handles to snap to the size of the text.



Verify alignment of text objects

Check the alignment of the field labels and their associated fields—they should both be aligned in the same direction. If you put a left-aligned label over a column of right-aligned numbers, for example, the report might look fine on your computer. But font substitution could cause field labels to shift when the file is opened on another computer. If a wider font is substituted on the second computer, your column heading will shift to the right. If a more narrow font is used, the text will display too far to the left.

Use merge fields

Use merge fields when mixing text and fields, for example, if you want to place a field in the middle of a sentence. Because character spacing may vary, avoid mixing text and fields.

Moving files between platforms

There are two ways to move files between Windows and Mac OS computers: by floppy disk or other removable media, or over a network.

There is no file conversion required when using FileMaker Pro files cross-platform, as the file format is the same for IBM-compatible and Mac OS-compatible computers. To get the file onto a floppy disk or removable media, however, requires file transfer, since the two computers format their media differently. To transfer files with floppy disks, you will need high-density disk drives.

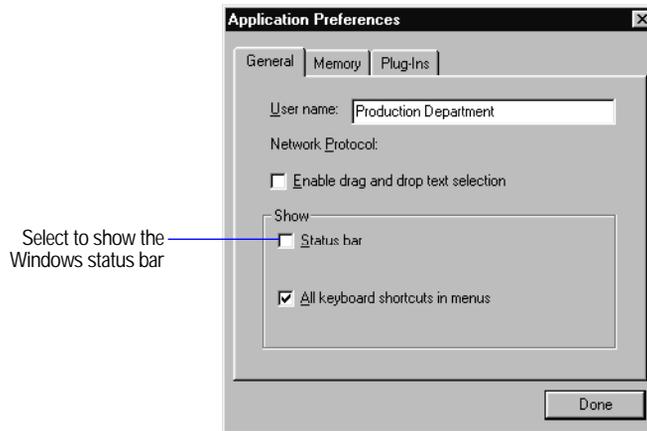
There are a variety of products on the market that enable one type of computer to read and write to a floppy disk formatted for another type of computer. PC Exchange version 2.2 is included with the Mac OS 8.1 system software and successfully transfers long Windows 95 and Windows NT filenames to the Mac OS.

Note If your file exchange utility doesn't handle long filenames between Windows 95 or Windows NT and the Mac OS, relationships and scripts may fail.

Contact your local software dealer for information on software that helps you transfer data across platforms.

Showing the status bar in Windows 95

FileMaker Pro for Windows has a general application preference to show the status bar, which allows a user to specify whether status bar information at the bottom of a window is visible or not.



Runtime Application Preferences dialog box

If your layouts are designed to take up most of the available screen space, end users can deselect this option.



Printing tips

Scripts that include Page Setup/Print Setup and Print are not 100% compatible cross-platform. The print steps in ScriptMaker rely upon the current printer driver in order to determine the paper sizes that are available, page orientation, and so on. Even if a computer using Windows and another with the Mac OS are connected to the same printer, the drivers themselves are significantly different—so FileMaker Pro is unable to restore page setup and print options cross-platform.

To work around this, you can:

- Create separate scripts for Windows and the Mac OS. Create the Windows script on a computer using Windows and the Mac OS script in the Mac OS. Use the Status (CurrentPlatform) function to determine whether to run the Windows or the Mac OS script. For more information, see “About the Status (CurrentPlatform) function” on page 3-15.
- In your Print script steps, be sure Perform Without Dialog is not selected. This will allow end users to change the setup options before they print.
- Specify document margins by selecting Layout Setup when in Layout mode.

QuickTime movies

To use a Mac OS QuickTime movie with FileMaker Pro for Windows, first save the movie in a format playable on non-Apple computers.

Use a QuickTime editing application or a utility such as MoviePlayer, which is available with Mac OS 8. For more information, visit the Apple web site: <http://www.apple.com>.

To save a movie in a format for other platforms:

1. In the Mac OS, start MoviePlayer.
2. Open the QuickTime movie you want to convert.
3. Choose Save As from the File menu.
4. Name the file in the Save As dialog box. If you will be distributing your solution to Windows 3.x users, use no more than 8 characters.
Be sure to add the .mov filename extension.
5. Select Make movie self-contained.
6. Select Make playable on non-Apple computers.

The first time you play a movie from the other platform (that is, a Windows movie in the Mac OS or a Mac OS movie in Windows), FileMaker Pro prompts you for the location of the movie. Keep the files in a common folder and avoid choosing the wrong file—this could lead to unexpected results.

Creating cross-platform scripts

Though most ScriptMaker steps work cross-platform, some steps rely upon or take advantage of platform-specific features. The following steps are platform-specific:

Windows-only script steps

- DDE Execute
- Send Message
- Insert Object
- Update Link

Mac OS-only script steps

- Send Apple Event
- Perform AppleScript
- Speak

Platform-specific script steps are skipped when run on a different platform. Also, platform-specific script steps are shown in italics when viewed on a different platform.

About the Status (CurrentPlatform) function

FileMaker Pro includes a status function that lets you determine the platform on which the solution is being run. This allows you to perform different script actions such as changing to a different layout or performing a platform-specific script step.

The Status (CurrentPlatform) function returns a 1 when run on a computer using the Mac OS and a 2 when run on a computer using Windows. Use this function with the If script step to perform different actions depending on the current platform.

```
If ["Status (CurrentPlatform)=1]
    Perform Script [Sub-scripts, "Print in Mac OS"]
Else
    Perform Script [Sub-scripts, "Print in Windows"]
End If
```

Converting and upgrading solution files

This section assumes you have developed a solution in the FileMaker Pro 3.0 or earlier Solutions Development Kit (SDK) and want to update your end users to FileMaker Pro 4.0 Developer Edition.

Here is an overview to updating your solution files:

1. Convert each solution file to FileMaker Pro 4.0.
2. If you want, update your solution files to take advantage of FileMaker Pro 4.0 features, such as the External Function plug-in, and Excel file importing.
3. If necessary, create scripts to import data from your end user's FileMaker Pro 3.0 or earlier solution files to the Developer Edition solution files.

Converting solution files

Because there was no file format change between FileMaker Pro 3.0 and 4.0, no conversion is necessary; FileMaker Pro 3.0 files can be bound by the FileMaker Pro 4.0 runtime application. Open FileMaker Pro 2.x and earlier solution files in FileMaker Pro 4.0. The original files will be renamed (“Filename Old”) and the converted files will keep the original name. If the solution files use relationships or external scripts, the converted files will continue to work together.

Importing to a new solution file

This section assumes your end users need to import data from their FileMaker Pro 3.0 and earlier solution files into the FileMaker Pro 4.0 solution files.

To import data from an earlier runtime solution file:

1. If necessary, convert copies of the solution files (FileMaker Pro 2.x files).

The converted filenames should be the same as the original filenames.

2. Add new features to the converted solution files.
3. Place the files to be converted in a folder inside the solution folder, for example `Solution folder\Old solution folder`.

If you have renamed the converted files, they can reside in the same folder as the old files.

4. Create scripts that import data from the original files into the updated solution files.

The original solution files will be converted when end users execute the script.

5. Test your scripts carefully. It is a good idea to use sample data to make sure the imports are working properly and data is going to the correct fields. Be sure to test the data on another machine to make sure the external scripts work.
6. Distribute the new solution files that contain the Import script.

The end users will run the script, which finds the original solution file, converts it, then imports the records into the new solution file. During the script, the runtime application temporarily converts a copy of the old file to FileMaker Pro 4.0, and deletes it when the converted file is closed. End users should delete the original solution files after making a backup copy.

If your end users only want to import a subset of the records from the original solution file, they can:

1. Convert the files manually.

The new solution files should contain a script that simply opens solution files.

2. In the newly converted solution files, perform find requests to locate the set of records they want imported.
3. In the new solution files, run the scripts you've provided that import data from the converted solution file to the new solution file.

Tip If you have scripts in the original solution files that help your end users reduce the found set, (for example, a script that Enters Find Mode and pauses) you can call that script before the import script step.

Keep the following considerations in mind:

- Provide instructions telling end users how to use your scripts to convert their solution files and import data into the new solution files.
- Use file size information to calculate how much disk space is necessary for the conversion to go smoothly. Keep in mind that a FileMaker Pro 2.x file that is opened via a script with the 4.0 runtime engine is copied, and then converted. For example, if a solution file is 2 MB, you will need at least 4 MB available to convert the solution.
- Set appropriate time expectations for your end users. If their solution files are large, the conversion and import process may be lengthy.

Chapter 4: *Distributing your bundled solution*

The final step in preparing your bundled solution is creating an installation procedure. You should:

- organize the files that end users need to use your solution
- choose how you will distribute your solution, for example on a CD-ROM, the World Wide Web, or on floppy disks
- provide end users with documentation for installing your solution

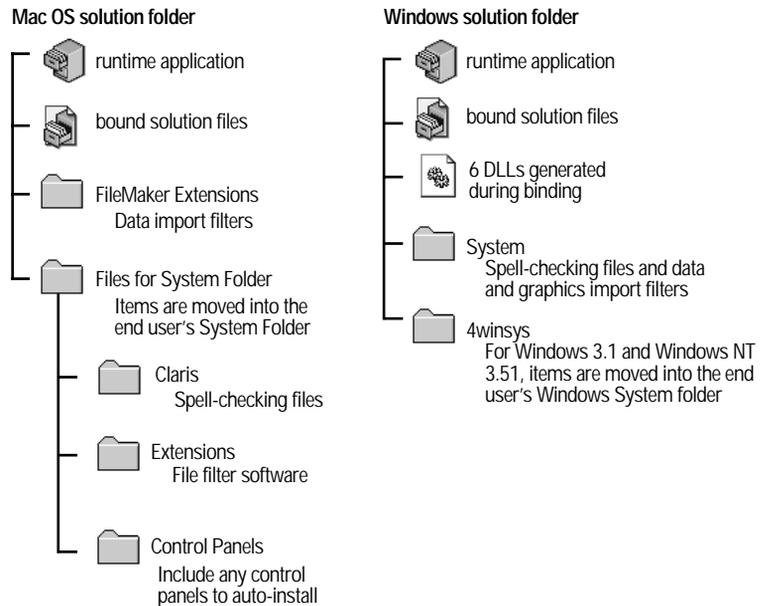
This chapter covers the above, as well as information on running, updating, and recovering solution files.

Organizing your solution files

Start by gathering all the files that are included in your bundled solution. These include the bound primary and auxiliary files, the runtime application, an end-user license agreement (see page 1-12) and, in Windows, six DLLs created during the binding process. The solution files are gathered in a folder created during the binding process (“<Primary filename> Solution Folder”).

Include any optional files that end users might need, such as spell-checking dictionaries or file filters, in the appropriate folder inside the solution folder. These optional files are located on the Developer Edition CD-ROM: Developer Edition CD\FileMaker Pro Binder\Runtime Files. See “Installing items from the Runtime Files folder” on page 4-3 for more information.

Important DLLs and optional files and folders inside the solution folder cannot be moved or renamed.



Sample Mac OS and Windows solution folders for distribution

Windows runtime solution files

You must copy the following files to the solution folder on the end user's hard disk. The minimum files required to run FileMaker Pro runtime solution files in Windows are:

- ClIngen.dll
- Clspell.dll
- Fmengine.dll
- Fmole.dll
- Fmtools.dll
- Fm<your extension>.dll
- the runtime application
- your bound solution files

The six DLLs are generated during the binding process and automatically moved into the solution folder.

The 4winsys folder

If your solution will be used in Windows 3.1 or Windows NT 3.51, you must include the 4winsys folder and its contents with your solution. It contains files that are necessary for the “3-D” look and feel. When the runtime solution is first opened, the appropriate file will be copied to the \Windows\System folder, and the 4winsys folder will be deleted. When the solution is opened in Windows 95, Windows 98, or Windows NT 4.0 or later, the 4winsys folder is unnecessary and is automatically deleted.

End users running Windows 3.1 or Windows for Workgroups 3.11 must install Win32s version 1.30c before opening the runtime application. The Developer Edition includes a copy for you to redistribute on the Developer Edition CD-ROM in the FileMaker Pro 4.0 folder. For more information on Win32s, see “Installing Win32s” on page 4-9.

Mac OS runtime solution files

For the Mac OS, the minimum files required to run FileMaker Pro runtime solution files are:

- the runtime application
- your bound solution files

Installing items from the Runtime Files folder

Items in the Runtime Files folder (Windows and the Mac OS) should be distributed along with your bound solution files and the runtime application to enable additional features, such as spell-checking and data and graphics importing.

To distribute the Runtime Files folder contents, drag the Runtime Files folder items from the CD-ROM into the root level of your solution folder. When the runtime application is first opened, these items will be moved to the appropriate locations on your end user’s hard disk.

Installing Runtime Files in Windows

Drag the Developer Edition CD\FileMaker Pro Binder\Runtime Files\System folder into the root level of your solution folder. The files should immediately work with the Runtime application on the end user’s hard disk.

In Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51, items in the `\Runtime Files\4winsys` folder will be moved to the `Windows\System` directory when the runtime application is first opened. The empty folder is then deleted. On Windows 98, Windows 95, and Windows NT 4.0, the folder is not necessary and is automatically deleted if you have provided it in your solution folder. For more information, see “Installing Win32s” on page 4-9.

Installing Runtime Files in the Mac OS

Drag the `\Runtime Files\FileMaker Extensions` folder into the root level of your solution folder.

Drag the `\Runtime Files\Files for System Folder` folder into the root level of your solution folder. When the Mac OS runtime application is opened, the Files for System Folder items are moved into the end user's System Folder.

Tip You can add folders and items inside Files for System Folder, and they will be moved to the appropriate location on the end user's hard disk. For example, if you create a Fonts folder in the Files for System Folder folder, and place a font in it, the font is moved into the Fonts folder on the end user's hard disk.

Contents of the Runtime Files folder

The following tables describe the files and folders inside the Runtime Files folder Developer Edition CD\`\FileMaker Pro Binder\Runtime Files`. Items in this folder should be distributed along with your bound solution files and the runtime application in the runtime solution folder.

Distribute the following files or folders with Windows runtime solutions.

File or folder	Description	Location on Developer Edition CD-ROM	Preferred location on end user's hard disk
System folder	Place in same folder as runtime application to enable data importing	FileMaker Pro Binder\ <code>Runtime Files</code>	Solution folder
Fmbmp32.flt	BMP graphic filter	FileMaker Pro Binder\ <code>Runtime Files\System</code>	Solution folder\ <code>System</code>
Fmcgm32.flt	CGM graphic filter	FileMaker Pro Binder\ <code>Runtime Files\System</code>	Solution folder\ <code>System</code>
Fmdrw32.flt	DRW graphic filter	FileMaker Pro Binder\ <code>Runtime Files\System</code>	Solution folder\ <code>System</code>

File or folder	Description	Location on Developer Edition CD-ROM	Preferred location on end user's hard disk
Fmgif32.flt	GIF graphic filter	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Fmhpgl32.flt	HPGL graphic filter	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Fmmac32.flt	MacPaint graphic filter	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Fmpcx32.flt	PCX graphic filter	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Fmpic32.flt	PIC graphic filter	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Fmsld32.flt	Lotus SLD graphic filter	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Fmtiff32.flt	TIFF graphic filter	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Fmwmf32.flt	Metafile graphic filter	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Cworks.imp	Distribute to import ClarisWorks databases	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Excel.imp	Distribute to import Excel files	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Dbf.imp	Distribute to import dBASE files	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Usenglish.mpr	The primary US English dictionary of over 100,000 words that FileMaker Pro uses to spell-check files	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Ukenglish.mpr	The primary British English dictionary of over 100,000 words that FileMaker Pro uses to spell-check files	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
User.usp	The dictionary that stores your customized list of spelling exceptions and additions to the User dictionary	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Fmatm16.dll	Support for Adobe Type Manager (Windows 95, 98, and NT only)	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
Fmatm32.dll	Support for Adobe Type Manager (Windows 95, 98, and NT only)	FileMaker Pro Binder\Runtime Files\System	Solution folder\System
4winsys	Distribute on Windows 3.1, Windows NT 3.51 or Windows for Workgroups 3.11	FileMaker Pro Binder\Runtime Files\System	Solution folder

File or folder	Description	Location on Developer Edition CD-ROM	Preferred location on end user's hard disk
Ctl3d32.dll	Provides "3-D" interface for Windows 3.1	FileMaker Pro Binder\Runtime Files\System	Solution folder\4winsys
Ctl3d32.nt	Provides "3-D" interface for Windows NT 3.51	FileMaker Pro Binder\Runtime Files\System	Solution folder\4winsys

Remember to distribute the Win32s 1.30c folder for your Windows 3.1 and Windows for Workgroups users. For more information, see "Installing Win32s" on page 4-9.

Distribute the following files or folders with Mac OS runtime solutions.

File or folder	Description	Location on Developer Edition CD-ROM	Preferred location on end user's hard disk
FileMaker Extensions folder	Place in same folder as runtime application to enable data importing	FileMaker Pro Binder: Runtime Files	Solution folder
DBF	Distribute to import dBASE files	FileMaker Pro Binder: Runtime Files: FileMaker Extensions folder	Solution folder: FileMaker Extensions
ClarisWorks	Distribute to import ClarisWorks databases	FileMaker Pro Binder: FileMaker Extensions folder	Solution folder: FileMaker Extensions
Excel	Distribute to import Excel files	FileMaker Pro Binder: Runtime Files: FileMaker Extensions folder	Solution folder: FileMaker Extensions
Files for System Folder	Distribute contents to enable spell-checking	FileMaker Pro Binder: Runtime Files folder	Solution folder
Claris Folder	Copy to end user's hard disk to enable spell-checking	FileMaker Pro Binder: Runtime Files: Files for System Folder folder	Solution folder: Files for System Folder
Usenglish.mpr	The primary US English dictionary of over 100,000 words that FileMaker Pro uses to spell-check files	FileMaker Pro Binder: Runtime Files: Files for System Folder: Claris Folder	Solution folder: Files for System Folder: Claris Folder
Ukenglish.mpr	The primary British English dictionary of over 100,000 words that FileMaker Pro uses to spell-check files	FileMaker Pro Binder: Runtime Files: Files for System Folder: Claris Folder	Solution folder: Files for System Folder: Claris Folder
User.upr	The dictionary that stores your customized list of spelling exceptions and additions to the User dictionary	FileMaker Pro Binder: Runtime Files: Files for System Folder: Claris Folder	Solution folder: Files for System Folder: Claris Folder
Extensions folder	Copy to end user's hard disk for Power PC support files	FileMaker Pro Binder: Runtime Files: Files for System Folder	Solution Folder: Files for System Folder
XTND Power Enabler	Enables graphics importing on Power Macintosh machines	FileMaker Pro Binder: Runtime Files: Files for System Folder: Extensions	Solution Folder: Files for System Folder: Extensions

File or folder	Description	Location on Developer Edition CD-ROM	Preferred location on end user's hard disk
Control Panels folder	Drop items in that you want auto-installed	FileMaker Pro Binder: Runtime Files: Files for System Folder:	Solution Folder: Files for System Folder
Claris Translators folder	Contains graphics import filters	FileMaker Pro Binder: Runtime Files: Files for System Folder: Claris Folder	Solution Folder: Files for System Folder: Claris Folder
EPSF PFLT	Enables importing of EPSF files	FileMaker Pro Binder: Runtime Files: Files for System Folder: Claris Folder: Claris Translators	Solution Folder: Files for System Folder: Claris Folder: Claris Translators
MacPaint 2.0	Enables importing of MacPaint 2.0 files	FileMaker Pro Binder: Runtime Files: Files for System Folder: Claris Folder: Claris Translators	Solution Folder: Files for System Folder: Claris Folder: Claris Translators
TIFF	Enables importing of TIFF files	FileMaker Pro Binder: Runtime Files: Files for System Folder: Claris Folder: Claris Translators	Solution Folder: Files for System Folder: Claris Folder: Claris Translators

Choosing the distribution method

After you have organized the files that comprise your solution, you need to decide how your end users will install them. If you plan to distribute your bundled solution on floppy disks, you will need to either use a compression utility or a custom installation application to compress and store the files on several disks. You can also distribute your bundled solution on a CD-ROM, over the World Wide Web, or other media, such as LS-120 or Zip disks.

Creating a custom installation

Because the runtime application does not fit on a floppy disk, you'll need to use software to compress your bound runtime solution. Configuring a custom installation application to automatically install runtime solution files may require more engineering than compression utilities, but usually offers more professional results.

The following trial installation applications are available in the Third-Party Tools folder on the Developer Tools CD-ROM:

- MindVision Installer VISE (Windows and Mac OS)
- InstallShield (Windows)
- StuffIt InstallerMaker (Mac OS)

Note Mention of third-party products is for informational use only and is not intended to be an endorsement. Refer to the product documentation for more information on installation applications.

Compressing your solution

If your runtime solution is small, you might consider a compression application rather than a custom installation application. To compress files, use a utility such as:

- WinZip for Windows by Nico Mak Computing, Inc. (www.winzip.com)
- StuffIt Deluxe for Mac OS (the StuffIt Expander decompression utility is available for Windows and Mac OS) by Aladdin Systems (www.aladdinsys.com)

Note Mention of third-party products is for informational use only and is not intended to be an endorsement.

Documenting your installation

You'll need to provide installation instructions to your end users. Keep the following in mind:

- Provide written instructions for copying or installing your solution to your end user's hard disk.
- Include software and instructions specifying how your end users will decompress your solution files.
- If your end users are installing your solution on Windows 3.1 or Windows for Workgroups 3.11, provide instructions for installing Win32s. See the next section, "Installing Win32s."
- If your solution might be run on a variety of operating systems, it is safest to include all optional runtime files.

For more solution documentation tips, see "Documenting your files" on page 3-4.

Installing Win32s

To run solution files on Windows 3.1 and Windows for Workgroups 3.11, your end users must first install Win32s 1.30c. Disk copies of Win32s 1.30c are available from the FileMaker Pro 4.0 folder on the Developer Edition CD-ROM.

Creating Win32s diskettes

To create Win32s diskettes:

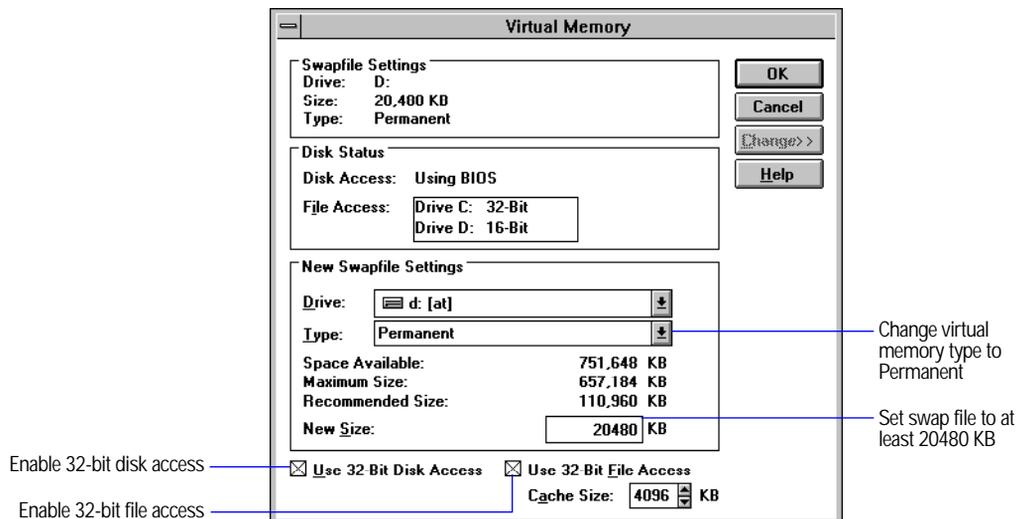
1. Format three 1.44 MB diskettes and name them Disk1, Disk2, and Disk3.
2. Drag the contents of the corresponding folders from the FileMaker Pro 4.0 folder on the Developer Edition CD-ROM onto the floppy disks.

Important Disk names, filenames, and paths cannot be modified or the Win32s installation will fail.

To install Win32s:

1. Open the 386 Enhanced control panel and change the virtual memory type to Permanent.
2. Make sure the swap file size is at least 20480 KB.
3. Select Use 32-Bit Disk Access and Use 32-Bit File Access, if available.

If any changes were made, they'll need to restart Windows before installing Win32s.



4. Install Win32s.

Instruct your end users to insert the Win32s Disk1. In the Program Manager, choose Run from the File menu and type `a:\setup.exe` from the Program Manager. Follow the Win32s setup instructions.

5. When the Win32s installation is complete, the FreeCell Setup dialog box appears.

FileMaker, Inc. recommends To verify that the Win32s software has been installed correctly, your end users should install the game FreeCell, included with Win32s version 1.30c.

6. Restart the machine before installing the bundled solution.

7. Make sure the 4winsys folder is in the solution folder on the end user's hard disk after the solution has been installed.

When installing a runtime solution on a computer with Windows 3.1, Windows NT 3.51, or Windows for Workgroups 3.11, be sure to distribute the 4winsys folder. This folder contains DLLs that the runtime application needs to display the application interface elements correctly.

Opening your runtime solution

The first time end users double-click the runtime application icon, the runtime application will auto-register and, if appropriate, move items into the operating system System folder. See “Installing items from the Runtime Files folder” on page 4-3 for more details.



In Windows, the three-character filename extension associated with the solution will not be registered by the operating system until the runtime application has been opened. If a solution file is double-clicked before the runtime application has registered the extension, the runtime application won't be found.



Your end users should start your solution by double-clicking the runtime application icon, not the primary file icon. Double-clicking the icons for the primary or auxiliary files might result in errors, depending on whether there are other copies of the runtime application on the hard disk. If your end users have more than one solution on their computers with the same three-character extension and they double-click the icon for the primary file, the most recently installed runtime application is opened, which may not be the correct application for your solution's primary file.

Each time the runtime application is opened, it looks for the primary file that has been bound to it. If the primary file can't be found, the end user is asked to locate the primary file.

Caution your end users that they should not rename the primary or auxiliary solution files. If they do, relationships and external scripts may not work properly. They can rename the runtime application after it has been installed.

Note When you make a change to your solution, you should make sure that your end users can import their data into your updated solution. Include a script attached to a button to make it easy for your end users to import their data into the new solution files. For more information, see “Converting and upgrading solution files” on page 3-16.

Updating a solution

To make feature enhancements, modifications, or add solution files, you must rebind and redistribute your runtime solution. If you are adding more files to a runtime solution, bind new files with the original key used when you bound the primary file.

Note To update a primary file, the filename and binding key must be the same as when you originally bound the file. If you change the name of the primary file, distribute a new version of the runtime application.

Updating the primary file

To update a primary file:

1. Open the original primary file in FileMaker Pro 4.0. Make the changes to the primary file.

As long as the key and the filename have not changed, you don't need to rebind the primary file.

2. Send your end users a copy of the new primary file.
3. If necessary, create scripts that import end users' existing data into the new primary file.

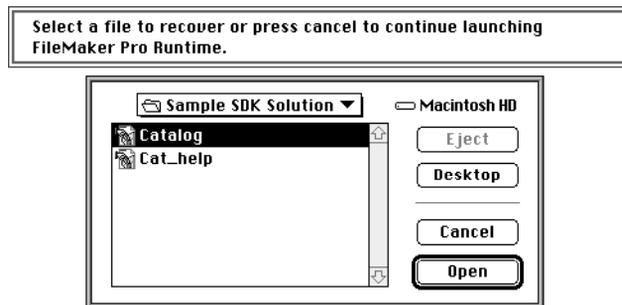
Adding new or changed auxiliary files

To add new or changed auxiliary files, rebind the solution files and generate a new runtime application. Use the same binding key that you used for the primary file. Remember that the binding key is case-sensitive in the Mac OS.

Recovering damaged files

Power failures, hardware problems, or other factors can damage a FileMaker Pro file. If your solution becomes damaged, your end user needs to recover the damaged file. When the runtime application discovers a damaged file, a dialog box appears, telling the end user to contact the developer.

Once you know which file is damaged, you can recover it; however, the Recover command does not appear in the File menu of the runtime application. To recover a damaged file in Windows, press Ctrl-Shift while double-clicking the runtime application icon. In the Mac OS, press Option-Command. Hold the keys down until you see the Recover File dialog box.



Recovering a runtime solution file

During the recovery process, the FileMaker Pro runtime application:

- creates a new file
- renames any damaged file by adding Old to the end of the filenames (for example, Contact Manager is renamed to Contact Manager Old).
- gives the repaired file the original name

If end users experience unusual behavior in the recovered files, they should revert to a backup copy that was made before the file became corrupt or contact you for technical assistance.

In your printed documentation, you should tell your end users what to do once a file has been recovered.

Tell your end users to:

1. Recover the damaged solution file.
2. Open the recovered solution file.
3. Save a compressed copy using the Save a Copy As command.
4. Give the compressed file the same name as the original filename.

For documentation ideas, see “Use custom Help script” on page 2-12 and “Documenting your files” on page 3-4.

Appendix A: Feature comparison—runtime application and FileMaker Pro

Key differences between the FileMaker Pro runtime application and FileMaker Pro 4.0 include:

- All the database design features have been removed or hidden in the runtime application.
- The runtime application cannot access files over a network.
- The runtime application cannot communicate with the FileMaker Web Companion. Hosting a database on the Web or communicating with a Java applet requires FileMaker Pro 4.0.
- Some menu items have been removed from the runtime application (see table below).
- The runtime application lets end users use your solution files without purchasing the FileMaker Pro application.

Menu command comparison

The following table shows which menu commands are available in the runtime application.

Menu	Menu Item	FileMaker Pro	Runtime application
File menu	New...	•	
	Open...	•	
	Close	•	
	Define Fields...	•	
	Define Relationships...	•	
	Define Value Lists...	•	
	Access Privileges	•	
	Change Password...	•	•
	Sharing	•	
	Page Setup or Print Setup...	•	•
	Print...	•	•
	Import/Export	•	•
	Save a Copy As...	•	•

Menu	Menu Item	FileMaker Pro	Runtime application
	Recover...	•	Ctrl-Shift in Windows; Option-Command in Mac OS
	Quit or Exit	•	•
Edit menu	Undo	•	•
	Cut	•	•
	Copy	•	•
	Paste	•	•
	Paste Formatted Text (Windows)	•	
	Paste As... (Windows)	•	Some options aren't available
	Clear	•	•
	Duplicate	•	
	Select All	•	•
	Paste Special	•	•
	Object (Windows)	•	•
	Spelling	•	•
	Preferences...	•	Some preferences aren't available
Mode menu	Browse	•	•
	Find	•	•
	Layout	•	
	Preview	•	•
	New Record	•	•
	Duplicate Record	•	•
	Delete Record	•	•
	Delete All	•	•
	Sort...	•	•
	Replace...	•	•
	Relookup	•	•
	Revert Record	•	•
Select menu	Find All	•	•
	Find Omitted	•	•
	Modify Last Find	•	•
	Perform Find	•	•

Menu	Menu Item	FileMaker Pro	Runtime application
	Omit	•	•
	Omit Multiple...	•	•
	View as Form	•	•
	View as List	•	•
Arrange menu	All menu items	•	
Format menu	Font	•	•
	Size	•	•
	Style	•	•
	Align Text	•	•
	Line Spacing	•	•
	Text Color	•	•
	Text...	•	•
	Text Ruler	•	•
Script menu	ScriptMaker...	•	
	<Script names>	•	•
Window menu	Hide Window (Mac OS)	•	•
	Tile (Windows)	•	•
	Cascade (Windows)	•	•
	Arrange Icons (Windows)	•	•
	<Names of open files>	•	•
Help menu (Windows)	FileMaker Help	•	Displays if custom Help script is specified
	FileMaker Help Topics	•	
	FileMaker Help Index	•	
	About FileMaker Pro/ FileMaker Pro Runtime	•	•
Help menu (Mac OS)	About Balloon Help	•	
	Show Balloons	•	
	FileMaker Pro Help	•	Displays if custom Help script is specified
	FileMaker Pro Help Topics	•	
	FileMaker Pro Help Index	•	
Apple menu (Mac OS)	About FileMaker Pro/ FileMaker Pro Runtime	•	•

Runtime application-specific features

There are some features that the runtime application supports that the FileMaker Pro 4.0 application does not, such as:

- running solutions in Kiosk mode
- specifying custom About, Help, and Script menu items

When a runtime solution file with these options is opened in FileMaker Pro 4.0, the file behaves as if these binding options were not selected.

Preferences differences

Not all of the Preferences options available in FileMaker Pro 4.0 are available in the runtime application.

The following Application Preferences aren't available in the runtime application:

- Show templates in New File dialog is not available
- Show recently opened files is not available
- None of the Layout application preferences are available (always locking layout tools, adding newly defined fields to current layout, creating rectangular or rounded buttons)
- None of the General Document Preferences are available (using smart quotes, storing compatible graphics, specifying a default password, executing scripts when opening or closing files, and switching to a particular layout when opening a file)

Important Specify the desired preferences before binding solution files.

Paste As... differences

Not all of the Paste As options available in FileMaker Pro 4.0 are available in the runtime application.

The following items aren't available in the Paste As submenu:

- Date Symbol
- Time Symbol
- User Name Symbol
- Page Number
- Record Number
- Merge Field...

Additional limitations

Other features have been removed from the runtime application. If your scripts contain any of the following items, the runtime application skips over them, and may not alert the end user. These items will not be executed.

- print script definitions (instead, the found set is printed)
- print field definitions (instead, the found set is printed)
- open Define Fields
- open Define Relationships
- open Define Value List
- open Sharing
- open ScriptMaker
- open Help (executes custom Help script specified during binding)
- Set Multi-User
- New (creates new files)
- perform external script (if the specified file has not been bound)
- script errors return different values with the runtime application

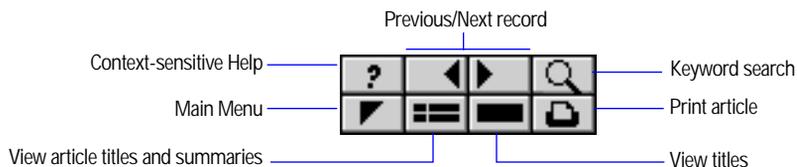
Appendix B: About the TechInfo database

The TechInfo database is a great resource for technical information about FileMaker, Inc. products. This FileMaker Pro database serves as a front-line resource for the company's support technicians as they field customer inquiries. It collects Q&As, tips, FAQs, bug reports, update notes, press releases, templates, and a host of other material valuable for the support professional. The TechInfo database is located in the Product & Support Information folder on the Developer Edition CD-ROM.

For late-breaking technical information, you can access a version of the TechInfo database from the support area of the FileMaker, Inc. web site (<http://www.filemaker.com>).

Navigating the TechInfo database

The navigation panel appears at the lower-left of each TechInfo screen. The buttons let you move between records in the found set, begin keyword searches, change the way you view the found set, print or fax a technical brief, return to the main menu, and summon online help.



Searching the TechInfo database

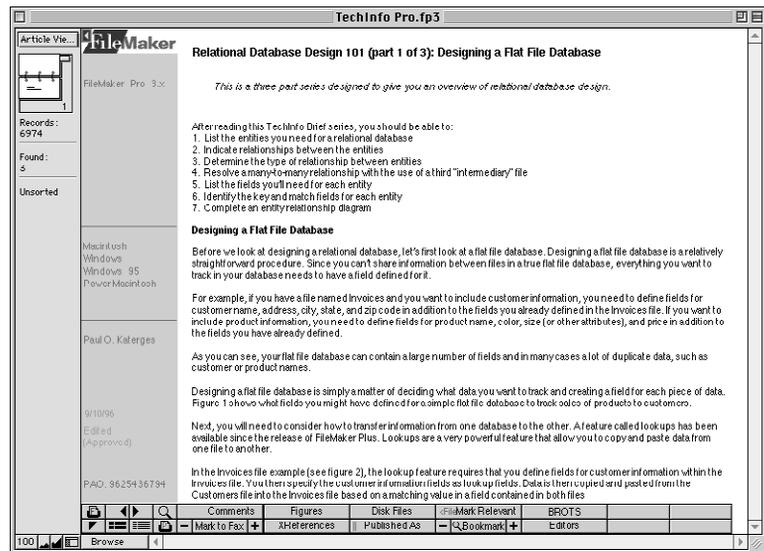
To locate information in the TechInfo database:

1. Click the magnifying glass button in the navigation panel to find records by keyword.
2. Articles that contain the words you selected are shown in a list. Browse through the list and select the article you want to read.
3. Click the Print button to print a specific article.

To print the found set of articles, modify the FileMaker Pro print settings in the Print dialog box.

Keep the following search tips in mind:

- If you don't find an article in the title or description fields, try a keyword search in the main text field.
- You can use the Find command in any layout.
- The Refind command is handy for specifying a new or slightly different search.
- Use the singular form, for example, “printer” instead of “printers.”
- Try roots of words. For example: “summar” instead of “summaries” or “summary.”
- Use generic terms. For example: “color postscript” instead of “QMS.”
- Try leaving the product field blank. Helpful information could be listed under another product or another version of a product.



TechInfo database article view

Context-sensitive help is available in most screens. If you get stuck, click the question mark button in the navigation panel.

Sample TechInfo article

Below is a sample TechInfo article that has been edited for this *Developer's Guide*. You can provide the backup script in the Script menu of your solution or create a button for your end users to back up their work.

How To Automatically Back up Your Database Upon Exit

If you are in need of an automated way to save back up copies of a FileMaker Pro 3.x or later database the following script will work well for you.

This script saves a copy of your database on the fifth close and every fifth close thereafter. To make the script work, you will need to define a number field called "Count Field". Define this script in all solution files that require backups.

In ScriptMaker, define the following script, named Backup:

```
Set Error Capture [On]
Show message ["Click to reset counter to zero"] {default
button = "No", 2nd button = "yes"}
If ["Status (CurrentMessageChoice)=2"]
    Set Field ["Count Field", "0"]
Else
    Set Field ["count field", "Count Field + 1"]
    If ["count field > 4"]
        Save a copy as ["Backup Copy"]
        Set Field ["Count Field", "0"]
    End if
End if
```

After adding all the steps above, click OK, then Done.

1. Choose Preferences from the Edit menu, then choose Document.
2. In the General tab (Windows) or the General pop-up menu (Mac OS), under When closing <solution filename>, select the Perform script checkbox.
3. Select the script Backup.
4. Click Done.
5. Close the solution file.

Once these steps are executed, you will have a script that will create a backup of your file every fifth time you close the file.

Other interesting TechInfo articles to search for:

- Ten Steps to Designing and Building FileMaker Pro Database Systems
- Relational Database Design 101
- Corrupt Files: Recovery and How to Avoid It
- Removing Corrupt Records From a File
- Merging Two Databases While Updating Records
- Hierarchical Menus in FileMaker Pro 3.0
- Troubleshooting Scripts: Hints and Tips
- How to Uninstall WIN32s From Your System
- FileMaker, Inc. Products and the Year 2000

Index

Numerics

3-character extension 2-14

4winsys folder
described 4-3, 4-4
folder contents 4-3

A

"About" layout
described 1-2, 2-10 to 2-12
example file 2-11
legal requirements 1-12, 3-6

Accelerator keys 2-14

Access privileges, removing access
to 2-10, 3-5

APIs

described 1-1, 1-3
external function plug-ins 1-4
Java support 1-7

Applets, Java 1-7

Artwork Sampler database 1-14

ASCII characters 3-9

Auxiliary files

binding 2-5
closing 3-2
described 2-3
opening 3-2
scripts for opening 3-2
updating 4-11

B

Backing up data 3-1, 3-4, B-3

Binder application

described 1-1
icon 2-3
installing 1-9 to 1-11
key features 1-2
running 2-3 to 2-19
Runtime Files folder 4-3
system requirements 1-9
terminology 2-2

Binding

auxiliary files 2-5

changing filename
extensions 2-17
described 2-1
files 2-3 to 2-19
files on network 2-1
options 2-6
preparing for 2-1, 3-1 to 3-5
primary file 2-3

Binding key

assigning 2-4
described 2-2
forgetting 2-20

Bitmaps 3-11

Bundled solutions

See also Solution files
described 2-2
distributing 4-1
filename extension 2-14
information for customers 2-11,
4-8
installing for end users 4-1
limiting access to 2-9
renaming 4-11
required files 4-2
starting 4-10
updating 3-16 to 3-17, 4-11

Buttons

creating for Kiosk mode 2-7
Main Menu 2-6, 2-8
sample 3-2

C

C/C++, writing plug-ins in 1-4

Calculations, external functions 1-4

Calendar Navigator database 1-14

CDML 1-7

CD-ROMs

contents 1-13
Developer Edition CD 1-13
Developer Tools CD 1-14

Character sets 3-9

Classes, Java 1-8

Closing solution files

described 3-2
in Kiosk mode 2-8

splash screen 2-10

Color palettes 3-10

Commands, available to end
users A-1

Compressing solution files 1-10, 4-8

Converting older files 3-16 to 3-17

Creating Dynamic Buttons

database 1-14

Creator code 1-5, 2-16

Cross-platform files

See also Mac OS, Windows

binding 2-20

binding key 2-5

design tips 3-8 to 3-15

filename extensions 2-15 to 2-17

filenames 2-4

Status (CurrentPlatform)

function 3-15

testing 3-7

Custom menus 1-2, 2-11 to 2-14

Customer support

for Developer Edition 1-12

for end users 1-12

Customized help, creating 2-12

D

Damaged files, recovering 4-12

Define Fields, removing access
to 2-10

Define Relationships, removing
access to 2-10

Design tips

for cross-platform files 3-8 to 3-15

for Kiosk mode 2-9

Desktop conflicts 2-15

Developer Edition

converting earlier files to 3-16 to
3-17

described 1-1

installing 1-9 to 1-11

registration and support 1-12

Developer Edition CD contents 1-13

Developer Tools CD contents 1-14

Dictionaries 4-5

Distributing bundled solutions 4-1
DLLs 4-2
Documentation
 for runtime solutions 3-4
 Help system 2-12
 installation instructions 4-8
 printed 2-13, 3-4

E

EFP Documentation.pdf 1-5
End users
 license agreement 1-12
 limiting access to solutions 2-9
Examples and Tools folder 1-14
Extensions, filename 2-14 to 2-16
External function plug-ins 1-4 to 1-6

F

.FP3 extension 2-15
Fields
 merge fields 3-12
 sizing for cross-platform 3-11
File recovery 4-12
FileMaker Pro
 Java Class Library 1-8
 layout tips 3-11
 Web Companion 1-3
 Web Companion and Java 1-8
FileMaker Pro Binder
 See Binder application
FileMaker Pro Developer Edition. *See*
 Developer Edition
FileMaker Pro Runtime
 See Runtime application
FileMaker Pro SDK, upgrading
 from 3-16
FileMaker Tools database 1-14
Filenames
 cross-platform 2-4, 3-13
 solution 2-4
 three-character extensions 2-14 to
 2-17
Files
 closing in Kiosk mode 2-8
 damaged 4-12
 distributing cross-platform 3-12
 filename extensions 2-14 to 2-17

importing new solution 3-16
 in Runtime Files folder,
 Mac OS 4-6
 in Runtime Files folder,
 Windows 4-4
 on CD-ROMs 1-13, 4-4 to 4-7
 opening and closing 3-2
 optional solution files 4-1
 renaming 4-11
 required solution files 4-2
 selecting auxiliary 2-5
 selecting primary 2-3
Filters, file import 4-4, 4-6
FMBanner applet 1-9
FMMemoPad applet 1-9
FMProProxy 1-8
FMProRequest 1-8
FMProResponse 1-8
Folders
 for solution files 2-17
 in Runtime Files folder,
 Mac OS 4-6
 in Runtime Files folder,
 Windows 4-4
 on CD-ROMs 1-13
 solution structure 4-2

Fonts
 character width 3-11
 cross-platform 3-8
 shadow style 3-9
Front-end, Java 1-7
Functions
 External 1-4
 Status (CurrentPlatform) 3-15

G

Graphic file formats 3-11

H

Help menu, custom 1-2
Help system for end users 2-12
HTML 1-7
HTTP 1-8

I

Importing
 data import filters 4-1, 4-4, 4-6
 solution files 3-16
Installers, for solution files 3-2, 4-7
Installing
 Developer Edition 1-9 to 1-11
 solutions for end users 4-1
 Win32s 4-9 to 4-10

J

Java
 classes 1-8 to 1-9
 described 1-7
 example applets 1-9
 using with FileMaker Pro 1-7

K

Key, binding
 assigning 2-4
 described 2-2
 forgetting 2-20
Kiosk mode
 and FileMaker Pro 4.0 1-11
 described 2-2
 designing for 2-9
 example file 2-7
 running solution in 2-7 to 2-9

L

Layouts
 Designing cross-platform 3-11
 the "About" layout 2-10, 3-6
 the Help layout 2-12
License agreement, end-user 1-12
Line spacing 3-11

M

Mac OS
 and Windows filename
 extensions 2-16
 Binder application
 requirements 1-10
 binding solutions for 2-1, 2-20
 creator code 1-5, 2-16
 designing solutions for 3-8 to 3-15

- FileMaker Extensions folder 4-4
- folder structure for solution 4-2
- plug-in names 1-5
- required solution files 4-3
- runtime application requirements 1-11
- Runtime Files folder structure 4-6
- system requirements 1-11
- Master access, removing 1-2, 2-9, 3-6
- Menu bars, hiding 2-7
- Menus, custom 1-2, 2-11 to 2-14
- Monitors 3-7, 3-10
- Movies, running cross-platform 3-14

N

- Naming runtime applications 2-4
- Networks 1-11

O

- Opening solution files 3-2

P

- Parameters, external function 1-4
- Password dialog box, suppressing 3-5
- Passwords
 - defining 3-5 to 3-7
 - limiting access 1-2
- PICT files 3-11
- Plug-ins
 - example files 1-6
 - external function 1-4 to 1-6
 - registering 1-5
 - support for 1-5
- Preferences
 - default password 3-5
 - external function plug-ins 1-4
 - status bar 3-13
 - unavailable in Runtime application A-4
- Primary file
 - binding 2-3
 - described 2-3
 - icon 2-19
 - locating 4-11
 - renaming 4-11

- specifying 2-3
- updating 4-11
- Printed documentation 2-13, 3-4
- Printing 3-3, 3-14
- Privileges, access
 - removing access to 2-10, 3-5

Q

- QuickTime movies 3-14

R

- Read-only files 1-10
- Recovering files 4-12
- Registering
 - Developer Edition 1-12
 - plug-ins 1-5
- Registry conflicts 2-15
- Relational Example database 1-14
- Renaming
 - files 3-4, 4-11
 - runtime application 4-11
 - Script menu 2-13
- Requirements
 - FileMaker Pro Binder 1-9
 - for Mac OS end users 1-11
 - for Windows end users 1-11
 - legal requirements 1-12
 - system 1-9
- Running your solution 4-10
- Runtime application
 - creating 2-1
 - described 1-1
 - distribution conditions 1-12
 - icon 2-19
 - installing for end users 4-1
 - menu commands available A-1
 - naming 2-4
 - networking solution files 1-11
 - renaming 4-11
 - running 4-10
- Runtime Files folder
 - contents of 4-4 to 4-7
 - described 4-1
 - System files 4-3
- Runtime Solution Example 1-14

S

- Sample
 - "About" layout 2-11
 - backup script B-3
 - buttons 3-2
 - end-user license 1-12
 - files 1-1
 - Help file 2-12, 3-2
 - installation applications 4-7
 - installer 3-2
 - Java applets 1-9
 - Kiosk example 2-7
 - plug-ins 1-6
- Script menu, custom 2-13
- ScriptMaker, removing access to 2-10
- Scripts
 - and long filenames 3-13
 - for "About" menu 2-10
 - for backing up data B-3
 - for Help system 2-12
 - for Kiosk mode 2-9
 - for Mac OS or Windows only 3-15
 - for printing 3-14
 - startup 2-6
 - startup script 2-8
 - unavailable in runtime application A-5
- SDK, converting solutions from 3-16 to 3-17
- Solution files
 - "About" layout 2-10
 - binding 2-1
 - compressing 1-10, 4-8
 - damaged 4-12
 - described 2-2
 - disable closing splash screen 2-10
 - distributing 4-1
 - documentation 3-4
 - filename extension 2-14
 - installing for end users 4-1
 - limiting access to 1-2, 2-9
 - minimum requirements 4-2, 4-3
 - naming 2-4
 - opening and closing 3-2
 - opening in FileMaker Pro 2-15
 - organizing 4-1
 - password-protection for 3-5
 - preparing for binding 2-1, 3-1 to 3-5
 - renaming 3-4, 4-11
 - running 4-10

- specifying solution folder 2-17
- testing 3-7
- updating 2-20, 3-16 to 3-17, 4-11
- Spell checking 4-5
- Starting Binder application 2-3
- Starting bundled solutions 4-11
- Startup script 2-6, 2-8
- Status bar, Windows 3-13
- Support
 - for Developer Edition 1-12
 - for end users 1-12
 - for external function plug-ins 1-5
- System files 4-3
- System requirements 1-9

T

- TechInfo database
 - described B-1
 - navigating B-1
 - sample article B-3
 - searching B-1
- Technical support
 - for Developer Edition 1-12
 - for end users 1-12
- Terminology 2-2
- Testing solution files 3-7
- Text
 - aligning in layouts 3-12
 - fonts 3-8, 3-12
- Three-character extension
 - assigning 2-14 to 2-17
 - described 2-2
- Tools 1-1, 1-14

U, V

- Updating
 - converting from SDK 3-16 to 3-17
 - primary file 4-11
 - solution files 2-20, 4-11
- URL Register database 1-14
- .USR extension 2-14, 2-16

W, X, Y, Z

- Web Companion 1-3
- Web site
 - Apple Computer, Inc. 1-5
 - FileMaker, Inc. 1-5
- Win32s, installing 4-9 to 4-10
- Windows
 - Binder application
 - requirements 1-9
 - binding solutions for 2-1, 2-20
 - designing solutions for 3-8 to 3-15
 - filename extensions 2-15
 - folder structure for solution 4-2
 - plug-in names 1-5
 - required solution files 4-2
 - runtime application
 - requirements 1-11
 - Runtime Files folder structure 4-4
 - showing status bar 3-13
 - system requirements 1-11
- Windows 3.1, files needed for 4-3