

# About This Book

---

This book, *Inside Macintosh: Files*, describes the parts of the Macintosh Operating System that allow you to manage files. It shows in detail how your application can handle the commands typically found in a File menu. It also provides a complete technical reference to the File Manager, the Standard File Package, the Alias Manager, and other file-related services provided by the system software.

If you are new to the Macintosh Operating System, you should begin with the chapter “Introduction to File Management.” This chapter describes the basic structure of Macintosh files and the hierarchical file system (HFS) used with Macintosh computers, and it shows how you can use the services provided by the Standard File Package, the File Manager, the Finder, and other system software components to create, open, update, and close files. Because this chapter is designed to be largely self-contained, the reference and summary sections in this chapter are subsets of the corresponding sections from the other chapters in this book.

Once you are familiar with basic file management on Macintosh computers, you might want to read other chapters in this book. The chapter “File Manager” describes how your application can manage shared files; search for specific files in a volume; obtain information about files, directories, and volumes; and perform other advanced operations. This chapter also describes how the File Manager organizes file and directory data on disk and in memory. Much of this information is of interest only to designers of very specialized applications or file-system utility programs.

If you want to customize the user interface for naming and identifying files, you need to read the chapter “Standard File Package.” It provides complete information on how to customize and display the dialog boxes that let the user specify the names and locations of files to be saved or opened.

If your application needs to keep track of particular files, directories, or volumes, you might want to use the Alias Manager. It helps you find objects in the file system, even if those objects have been moved or renamed. See the chapter “Alias Manager” for complete details.

The chapter “Disk Initialization Manager” shows how you can initialize disks and erase the contents of previously initialized disks. The Disk Initialization Manager provides a routine that allows you to present the standard user interface for initializing and naming disks. Most applications should call that routine whenever they receive a disk-inserted event and the inserted disk is invalid.

## Format of a Typical Chapter

---

Almost all chapters in this book follow a standard structure. For example, the chapter “Standard File Package” contains these sections:

- “About the Standard File Package.” This section provides an overview of the features provided by the Standard File Package.
- “Using the Standard File Package.” This section describes the tasks you can accomplish using the Standard File Package. It describes how to use the most common routines, gives related user interface information, provides code samples, and supplies additional information.
- “Standard File Package Reference.” This section provides a complete reference to the Standard File Package by describing the data structures and routines that it uses. Each routine description also follows a standard format, which gives the routine declaration and a description of every parameter of the routine. Some routine descriptions also give additional descriptive information, such as assembly-language information or result codes.
- “Summary of the Standard File Package.” This section provides the Standard File Package’s Pascal interface, as well as the C interface, for the constants, data structures, routines, and result codes associated with the Standard File Package. It also includes relevant assembly-language interface information.

Some chapters contain additional main sections that provide more detailed discussions of certain topics. For example, the chapter “File Manager” contains the section “Identifying Files, Directories, and Volumes,” which describes the many ways to identify objects in the file system. That chapter also contains the two advanced sections “Data Organization on Volumes” and “Data Organization in Memory.”

## Conventions Used in This Book

---

*Inside Macintosh* uses various conventions to present information. Words that require special treatment appear in specific fonts or font styles. Certain information, such as parameter blocks, use special formats so that you can scan them quickly.

### Special Fonts

---

All code listings, reserved words, and the names of actual data structures, constants, fields, parameters, and routines are shown in Courier (`this is Courier`).

Words that appear in **boldface** are key terms or concepts and are defined in the Glossary.

## Types of Notes

---

There are several types of notes used in this book.

### Note

A note like this contains information that is interesting but possibly not essential to an understanding of the main text. (An example appears on page 1-6.) ♦

### IMPORTANT

A note like this contains information that is essential for an understanding of the main text. (An example appears on page 1-6.) ▲

### ▲ WARNING

Warnings like this indicate potential problems that you should be aware of as you design your application. Failure to heed these warnings could result in system crashes or loss of data. (An example appears on page 1-46.) ▲

## Assembly-Language Information

---

*Inside Macintosh* provides information about the registers for specific routines like this:

### Registers on entry

A0            Contents of register A0 on entry

### Registers on exit

D0            Contents of register D0 on exit

In addition, *Inside Macintosh* presents information about the fields of a parameter block in this format:

### Parameter block

↔	inAndOut	Integer	Input/output parameter.
←	output1	Ptr	Output parameter.
→	input1	Ptr	Input parameter.

The arrow in the far left column indicates whether the field is an input parameter, output parameter, or both. You must supply values for all input parameters and input/output parameters. The routine returns values in output parameters and input/output parameters.

The second column shows the field name as defined in the MPW Pascal interface files; the third column indicates the Pascal data type of that field. The fourth column provides a brief description of the use of the field. For a complete description of each field, see the discussion that follows the parameter block or the description of the parameter block in the reference section of the chapter.

## Development Environment

---

The system software routines described in this book are available using Pascal, C, or assembly-language interfaces. How you access these routines depends on the development environment you are using. This book shows system software routines in their Pascal interface using the Macintosh Programmer's Workshop (MPW).

All code listings in this book are shown in Pascal. They show methods of using various routines and illustrate techniques for accomplishing particular tasks. All code listings have been compiled and, in most cases, tested. However, Apple Computer does not intend that you use these code samples in your application.

This book occasionally uses *SurfDraw* as the name of a sample application for illustrative purposes; this is not an actual product of Apple Computer, Inc.

APDA, Apple's source for developer tools, offers worldwide access to a broad range of programming products, resources, and information for anyone developing on Apple platforms. You'll find the most current versions of Apple and third-party development tools, debuggers, compilers, languages, and technical references for all Apple platforms. To establish an APDA account, obtain additional ordering information, or find out about site licensing and developer training programs, contact

APDA

Apple Computer, Inc.

20525 Mariani Avenue, M/S 33-G

Cupertino, CA 95014-6299

Telephone: 800-282-2732 (United States)  
800-637-0029 (Canada)  
800-562-3910 (elsewhere in the world)

Fax: 408-562-3971

Telex: 171-576

If you provide commercial products and services, call 408-974-4897 for information on the developer support programs available from Apple.

For information on registering signatures, file types, Apple events, and other technical information, contact

Macintosh Developer Technical Support

Apple Computer, Inc.

20525 Mariani Avenue, M/S 75-3T

Cupertino, CA 95014-6299