

INSIDE MACINTOSH

Macintosh Toolbox Essentials



Addison-Wesley Publishing Company

Reading, Massachusetts Menlo Park, California New York
Don Mills, Ontario Wokingham, England Amsterdam Bonn
Sydney Singapore Tokyo Madrid San Juan
Paris Seoul Milan Mexico City Taipei

Apple Computer, Inc.
© 1992, Apple Computer, Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Computer, Inc. Printed in the United States of America.

No licenses, express or implied, are granted with respect to any of the technology described in this book. Apple retains all intellectual property rights associated with the technology described in this book. This book is intended to assist application developers to develop applications only for Apple Macintosh computers.

Apple Computer, Inc.
20525 Mariani Avenue
Cupertino, CA 95014
408-996-1010

Apple, the Apple logo, APDA, AppleShare, AppleTalk, A/UX, EtherTalk, LaserWriter, Macintosh, MPW, and MultiFinder and are trademarks of Apple Computer, Inc., registered in the United States and other countries.

Apple Desktop Bus, Balloon Help, BalloonWriter, Chicago, Finder, Geneva, KanjiTalk, Monaco, New York, QuickDraw, QuickTime, ResEdit, System 7, and TrueType are trademarks of Apple Computer, Inc.

Adobe Illustrator and PostScript are trademarks of Adobe Systems Incorporated, which may be registered in certain jurisdictions.

AGFA is a trademark of Agfa-Gevaert.

FrameMaker is a registered trademark of Frame Technology Corporation.

Helvetica, Palatino, andTimes are registered trademarks of Linotype Company.

ITC Zapf Dingbats is a registered trademark of International Typeface Corporation.

Simultaneously published in the United States and Canada.

LIMITED WARRANTY ON MEDIA AND REPLACEMENT

ALL IMPLIED WARRANTIES ON THIS MANUAL, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN DURATION TO NINETY (90) DAYS FROM THE DATE OF THE ORIGINAL RETAIL PURCHASE OF THIS PRODUCT.

Even though Apple has reviewed this manual, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS MANUAL, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS MANUAL IS SOLD "AS IS," AND YOU, THE PURCHASER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS MANUAL, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

ISBN 0-201-63243-8
1 2 3 4 5 6 7 8 9-MU-9695949392
First Printing, October 1992

The paper used in this book meets the EPA standards for recycled fiber.

Contents

	Figures, Tables, and Listings	xv
Preface	About This Book	xxv
	Format of a Typical Chapter	xxvi
	Conventions Used in This Book	xxvi
	Special Fonts	xxvii
	Types of Notes	xxvii
	Empty Strings	xxvii
	Assembly-Language Information	xxvii
	The Development Environment	xxviii
Chapter 1	Introduction to the Macintosh Toolbox	1-1
	Overview of the Macintosh Toolbox	1-4
	Events	1-5
	Menus	1-6
	Windows	1-6
	Controls	1-7
	Alert Boxes and Dialog Boxes	1-8
	Icons and Other Interactions With the Finder	1-10
	Resources	1-11
	Help Balloons	1-14
	Copy and Paste	1-14
	Related System Software Features	1-14
	Drawing on the Screen	1-14
	Handling Text	1-14
	Managing Files	1-15
	Allocating Memory and Launching Processes	1-15
	Creating Publishers and Subscribers	1-15
	Communicating With Other Applications	1-16
	Designing Your Application	1-16
Chapter 2	Event Manager	2-1
	Introduction to Events	2-4
	Low-Level Events	2-8
	Operating-System Events	2-10
	High-Level Events	2-13
	Priority of Events	2-15

Switching Contexts	2-15
About the Event Manager	2-16
Using the Event Manager	2-17
Obtaining Information About Events	2-18
Processing Events	2-21
Using the WaitNextEvent Function	2-22
Writing an Event Loop	2-24
Setting the Event Mask	2-26
Handling Events in a Dialog Box	2-29
Creating a Size Resource	2-30
Handling Low-Level Events	2-32
Responding to Mouse Events	2-33
Responding to Keyboard Events	2-38
Scanning for a Cancel Event	2-46
Responding to Update Events	2-47
Responding to Activate Events	2-50
Responding to Disk-Inserted Events	2-55
Responding to Null Events	2-57
Handling Operating-System Events	2-58
Responding to Suspend and Resume Events	2-60
Responding to Mouse-Moved Events	2-62
Handling High-Level Events	2-67
Responding to Events From Other Applications	2-69
Searching for a Specific High-Level Event	2-71
Determining the Sender of a High-Level Event	2-72
Sending High-Level Events	2-73
Requesting Return Receipts	2-77
Handling Apple Events	2-78
Event Manager Reference	2-78
Data Structures	2-79
The Event Record	2-79
The Target ID Record	2-81
The High-Level Event Message Record	2-82
The Event Queue	2-83
Event Manager Routines	2-84
Receiving Events	2-84
Sending Events	2-100
Converting Process Serial Numbers and Port Names	2-105
Reading the Mouse	2-108
Reading the Keyboard	2-110
Getting Timing Information	2-112
Application-Defined Routine	2-114
Filter Function for Searching the High-Level Event Queue	2-114
Resource	2-115
The Size Resource	2-115
Summary of the Event Manager	2-120
Pascal Summary	2-120

Constants	2-120
Data Types	2-122
Event Manager Routines	2-123
Application-Defined Routine	2-124
C Summary	2-125
Constants	2-125
Data Types	2-127
Event Manager Routines	2-128
Application-Defined Routine	2-129
Assembly-Language Summary	2-130
Data Structures	2-130
Trap Macros	2-130
Global Variables	2-131
Result Codes	2-132

Chapter 3

Menu Manager 3-1

Introduction to Menus	3-5
Menu and Menu Bar Definition Routines	3-9
The Menu Bar	3-9
Menus	3-11
Menu Items	3-12
Groups of Menu Items	3-14
Keyboard Equivalents for Menu Commands	3-16
Menus Added Automatically by the Menu Manager	3-19
The Apple Menu	3-20
The File Menu	3-22
The Edit Menu	3-24
The Font Menu	3-26
The Size Menu	3-27
The Help Menu	3-29
The Keyboard Menu	3-32
The Application Menu	3-33
Pop-Up Menus	3-33
Hierarchical Menus	3-38
About the Menu Manager	3-39
How the Menu Manager Maintains Information About Menus	3-40
How the Menu Manager Maintains Information About an Application's Menu Bar	3-40
Using the Menu Manager	3-41
Creating a Menu	3-42
Creating a Menu Resource	3-43
Creating a Menu Bar Resource	3-49
Setting Up Your Application's Menu Bar	3-50
Creating a Hierarchical Menu	3-53
Creating a Pop-Up Menu	3-56

Changing the Appearance of Items in a Menu	3-57
Enabling and Disabling Menu Items	3-58
Changing the Text of an Item	3-59
Changing the Font Style of Menu Items	3-60
Changing the Mark of Menu Items	3-61
Changing the Icon or Script Code of Menu Items	3-62
Adding Items to a Menu	3-64
Adding Items to the Help Menu	3-67
Adding Items to the Apple Menu	3-68
Adding Fonts to a Menu	3-69
Handling User Choice of a Menu Command	3-70
Handling Mouse-Down Events in the Menu Bar	3-72
Adjusting the Menus of an Application	3-73
Determining if the User Chose a Keyboard Equivalent	3-77
Responding When the User Chooses a Menu Item	3-78
Handling the Apple Menu	3-80
Handling the Help Menu	3-81
Handling a Size Menu	3-82
Accessing Menus From a Dialog Box	3-84
Writing Your Own Menu Definition Procedure	3-87
Calculating the Dimensions of a Menu	3-89
Drawing Menu Items in a Menu	3-90
Determining Whether the Cursor Is in an Enabled Menu Item	3-92
Menu Manager Reference	3-95
Data Structures	3-95
The Menu Record	3-95
The Menu List	3-97
The Menu Color Information Table Record	3-98
Menu Manager Routines	3-102
Initializing the Menu Manager	3-103
Creating Menus	3-105
Adding Menus to and Removing Menus From the Current Menu List	3-108
Getting a Menu Bar Description From an 'MBAR' Resource	3-110
Getting and Setting the Menu Bar	3-112
Drawing the Menu Bar	3-113
Responding to the User's Choice of a Menu Command	3-114
Getting a Handle to a Menu Record	3-122
Adding and Deleting Menu Items	3-124
Getting and Setting the Appearance of Menu Items	3-130
Disposing of Menus	3-140
Counting the Items in a Menu	3-140
Highlighting the Menu Bar	3-141
Recalculating Menu Dimensions	3-142
Managing Entries in the Menu Color Information Table	3-143
Application-Defined Routine	3-148
The Menu Definition Procedure	3-148

Resources	3-151
The Menu Resource	3-151
The Menu Bar Resource	3-155
The Menu Color Information Table Resource	3-155
The Menu Definition Procedure Resource	3-157
Summary of the Menu Manager	3-158
Pascal Summary	3-158
Constants	3-158
Data Types	3-158
Menu Manager Routines	3-159
Application-Defined Routine	3-162
C Summary	3-162
Constants	3-162
Data Types	3-163
Menu Manager Routines	3-164
Application-Defined Routine	3-166
Assembly-Language Summary	3-167
Data Structures	3-167
Global Variables	3-167
Result Codes	3-167

Chapter 4	Window Manager	4-1
-----------	-----------------------	-----

Introduction to Windows	4-4
Active and Inactive Windows	4-6
Types of Windows	4-8
Window Regions	4-12
Dialog Boxes and Alert Boxes	4-13
Controls	4-14
Windows on the Desktop	4-15
About the Window Manager	4-16
Graphics Ports	4-17
Window Records	4-19
Color Windows	4-20
Events in Windows	4-21
Using the Window Manager	4-22
Managing Multiple Windows	4-23
Creating a Window	4-25
Defining a Window Resource	4-25
Creating a Window From a Resource	4-27
Positioning a Document Window on the Desktop	4-30
Drawing the Window Contents	4-39
Updating the Content Region	4-40
Maintaining the Update Region	4-41
Handling Events in Windows	4-41
Handling Mouse Events in Windows	4-42

Handling Keyboard Events in Windows	4-47	
Handling Update Events	4-48	
Handling Activate Events	4-50	
Moving a Window	4-53	
Zooming a Window	4-53	
Resizing a Window	4-57	
Closing a Window	4-60	
Hiding and Showing a Window	4-62	
Window Manager Reference	4-64	
Data Structures	4-65	
The Color Window Record	4-65	
The Window Record	4-69	
The Window State Data Record	4-70	
The Window Color Table Record	4-71	
The Auxiliary Window Record	4-73	
The Window List	4-74	
Window Manager Routines	4-74	
Initializing the Window Manager	4-74	
Creating Windows	4-75	
Naming Windows	4-85	
Displaying Windows	4-86	
Retrieving Window Information	4-91	
Moving Windows	4-94	
Resizing Windows	4-99	
Zooming Windows	4-101	
Closing and Deallocating Windows	4-103	
Maintaining the Update Region	4-106	
Setting and Retrieving Other Window Characteristics	4-109	
Manipulating the Desktop	4-112	
Manipulating Window Color Information	4-114	
Low-Level Routines	4-116	
Application-Defined Routine	4-120	
The Window Definition Function	4-120	
Resources	4-124	
The Window Resource	4-124	
The Window Definition Function Resource	4-127	
The Window Color Table Resource	4-127	
Summary of the Window Manager	4-130	
Pascal Summary	4-130	
Constants	4-130	
Data Types	4-132	
Window Manager Routines	4-134	
Application-Defined Routine	4-136	
C Summary	4-137	
Constants	4-137	
Data Types	4-139	

Window Manager Routines	4-140
Application-Defined Routine	4-143
Assembly-Language Summary	4-144
Data Types	4-144
Global Variables	4-145

Chapter 5 Control Manager 5-1

Introduction to Controls	5-4
Buttons	5-5
Checkboxes	5-5
Radio Buttons	5-6
Pop-Up Menus	5-6
Scroll Bars	5-7
Other Controls	5-11
Active and Inactive Controls	5-11
The Control Definition Function	5-14
About the Control Manager	5-14
Using the Control Manager	5-15
Creating and Displaying a Control	5-15
Creating a Button, Checkbox, or Radio Button	5-17
Creating Scroll Bars	5-21
Creating a Pop-Up Menu	5-25
Updating a Control	5-29
Responding to Mouse Events in a Control	5-30
Determining a Mouse-Down Event in a Control	5-31
Tracking the Cursor in a Control	5-35
Determining and Changing Control Settings	5-37
Scrolling Through a Document	5-43
Scrolling in Response to Events in the Scroll Box	5-53
Scrolling in Response to Events in Scroll Arrows and Gray Areas	5-57
Drawing a Scrolled Document Inside a Window	5-62
Moving and Resizing Scroll Bars	5-65
Defining Your Own Control Definition Function	5-71
Control Manager Reference	5-72
Data Structures	5-72
The Control Record	5-73
The Auxiliary Control Record	5-76
The Pop-Up Menu Private Data Record	5-77
The Control Color Table Record	5-77
Control Manager Routines	5-80
Creating Controls	5-81
Drawing Controls	5-85
Handling Mouse Events in Controls	5-88
Changing Control Settings and Display	5-93
Determining Control Values	5-102
Removing Controls	5-108

Application-Defined Routines	5-109	
Defining Your Own Control Definition Function		5-109
Defining Your Own Action Procedures	5-115	
Resources	5-117	
The Control Resource	5-118	
The Control Color Table Resource	5-121	
The Control Definition Function	5-123	
Summary of the Control Manager	5-124	
Pascal Summary	5-124	
Constants	5-124	
Data Types	5-126	
Control Manager Routines	5-127	
Application-Defined Routines	5-129	
C Summary	5-129	
Constants	5-129	
Data Types	5-131	
Control Manager Routines	5-132	
Application-Defined Routines	5-134	
Assembly-Language Summary	5-134	
Data Structures	5-134	
Global Variables	5-135	

Chapter 6

Dialog Manager 6-1

Introduction to Alerts and Dialog Boxes	6-6	
Types of Alerts	6-8	
Types of Dialog Boxes	6-9	
Modal Dialog Boxes	6-10	
Movable Modal Dialog Boxes	6-11	
Modeless Dialog Boxes	6-12	
Items in Alert and Dialog Boxes	6-13	
Events in Alert and Dialog Boxes	6-14	
Alert Boxes, Dialog Boxes, and the Window Manager	6-15	
About the Dialog Manager	6-16	
Using the Dialog Manager	6-17	
Creating Alert Sounds and Alert Boxes	6-18	
Creating Dialog Boxes	6-23	
Providing Items for Alert and Dialog Boxes	6-26	
Item Types	6-30	
Display Rectangles	6-32	
Enabled and Disabled Items	6-36	
Resource IDs for Items	6-36	
Titles for Buttons, Checkboxes, and Radio Buttons	6-37	
Text Strings for Static Text and Editable Text Items	6-40	
Pop-Up Menus as Items	6-42	
Keyboard Navigation Among Items	6-44	

Manipulating Items	6-44
Changing Static Text	6-46
Getting Text From Editable Text Items	6-48
Adding Items to an Existing Dialog Box	6-51
Using an Application-Defined Item to Draw the Bold Outline for a Default Button	6-56
Displaying Alert and Dialog Boxes	6-61
Positioning Alert and Dialog Boxes	6-62
Deactivating Windows Behind Alert and Modal Dialog Boxes	6-64
Displaying Modeless Dialog Boxes	6-66
Adjusting Menus for Modal Dialog Boxes	6-68
Adjusting Menus for Movable Modal and Modeless Dialog Boxes	6-73
Displaying Multiple Alert and Dialog Boxes	6-74
Displaying Alert and Dialog Boxes From the Background	6-74
Including Color in Your Alert and Dialog Boxes	6-75
Handling Events in Alert and Dialog Boxes	6-77
Responding to Events in Controls	6-78
Responding to Events in Editable Text Items	6-79
Responding to Events in Alert Boxes	6-81
Responding to Events in Modal Dialog Boxes	6-82
Writing an Event Filter Function for Alert and Modal Dialog Boxes	6-86
Responding to Mouse Events in Modeless and Movable Modal Dialog Boxes	6-89
Responding to Keyboard Events in Modeless and Movable Modal Dialog Boxes	6-94
Responding to Activate and Update Events in Modeless and Movable Modal Dialog Boxes	6-97
Closing Dialog Boxes	6-100
Dialog Manager Reference	6-101
Data Structure	6-101
The Dialog Record	6-101
Dialog Manager Routines	6-102
Initializing the Dialog Manager	6-103
Creating Alerts	6-105
Creating and Disposing of Dialog Boxes	6-113
Manipulating Items in Alert and Dialog Boxes	6-120
Handling Text in Alert and Dialog Boxes	6-129
Handling Events in Dialog Boxes	6-135
Application-Defined Routines	6-143
Resources	6-147
The Dialog Resource	6-148
The Alert Resource	6-150
The Item List Resource	6-151
The Dialog Color Table Resource	6-156
The Alert Color Table Resource	6-157
The Item Color Table Resource	6-158

Summary of the Dialog Manager	6-165
Pascal Summary	6-165
Constants	6-165
Data Types	6-166
Dialog Manager Routines	6-166
Application-Defined Routines	6-168
C Summary	6-168
Constants	6-168
Data Types	6-169
Dialog Manager Routines	6-170
Application-Defined Routines	6-172
Assembly-Language Summary	6-172
Data Structures	6-172
Global Variables	6-172

Chapter 7

Finder Interface 7-1

Introduction to the Finder Interface	7-3
About the Finder Interface	7-6
Using the Finder Interface	7-6
Giving a Signature to Your Application and a Creator and a File Type to Your Documents	7-8
Creating Icons for the Finder	7-11
Creating Customized Document Icons	7-17
Creating File Reference Resources	7-18
Creating a Bundle Resource	7-20
How and When the Finder Launches Your Application	7-25
Displaying Messages When the Finder Can't Find Your Application	7-27
Providing Version Resources	7-31
Using Finder Information in the Catalog File	7-32
Supporting Stationery Pads	7-34
Distributing Fonts, Sounds, and Other Movable Resources	7-36
Providing Balloon Help for Nondocument Icons	7-38
Using Aliases	7-39
Using the System Folder and Its Related Directories	7-41
The Desktop Database	7-45
Finder Interface Reference	7-46
Data Structures	7-46
File Information Record	7-47
Extended File Information Record	7-49
Directory Information Record	7-50
Extended Directory Information Record	7-50
Routines	7-51
Resolving Alias Files	7-51
Finding Directories	7-53

Resources	7-56
The Signature Resource	7-57
The Icon List Resource	7-57
The Small Icon List Resource	7-58
The Large 4-Bit Color Icon Resource	7-59
The Small 4-Bit Color Icon Resource	7-60
The Large 8-Bit Color Icon Resource	7-61
The Small 8-Bit Color Icon Resource	7-62
The Icon Resource	7-63
The Color Icon Resource	7-64
The File Reference Resource	7-64
The Bundle Resource	7-65
The Missing-Application Name String	7-68
The Application-Missing Message String	7-68
The Version Resource	7-69
Summary of the Finder Interface	7-71
Pascal Summary	7-71
Constants	7-71
Data Types	7-73
Routines	7-74
C Summary	7-74
Constants	7-74
Data Types	7-76
Routines	7-77
Assembly-Language Summary	7-77
Data Structures	7-77
Result Codes	7-78

Glossary GL-1

Index IN-1

Figures, Tables, and Listings

Color Plates

Color plates are immediately preceding the title page.

- Color Plate 1** A color window
- Color Plate 2** A color alert box
- Color Plate 3** An icon family
- Color Plate 4** Document icons

Chapter 1

Introduction to the Macintosh Toolbox 1-1

- Figure 1-1** The SurfWriter application with multiple windows on the desktop 1-3
- Figure 1-2** A typical window 1-6
- Figure 1-3** Common controls 1-7
- Figure 1-4** An alert box 1-8
- Figure 1-5** Modal, movable modal, and modeless dialog boxes 1-9

Chapter 2

Event Manager 2-1

- Figure 2-1** Sources of events sent to your application 2-6
- Figure 2-2** Low-level events 2-10
- Figure 2-3** Operating-system events 2-11
- Figure 2-4** High-level events 2-14
- Figure 2-5** The `modifiers` field of the event record 2-20
- Figure 2-6** The event mask 2-27
- Figure 2-7** The `message` field of the event record for keyboard events 2-40
- Figure 2-8** Keyboard translation 2-41
- Figure 2-9** Virtual key codes for the Apple Keyboard II, ISO layout 2-42
- Figure 2-10** Virtual key codes for the Apple Extended Keyboard II 2-43
- Figure 2-11** Responding to an update event for a window 2-49
- Figure 2-12** Responding to activate events for a window 2-52
- Figure 2-13** The standard arrow cursor 2-63
- Figure 2-14** The I-beam, crosshairs, plus sign, and wristwatch cursors 2-63
- Figure 2-15** The arrow region and the I-beam region 2-64
- Figure 2-16** Changing the cursor from the I-beam cursor to the arrow cursor 2-65
- Figure 2-17** Structure of the `KeyTranslate` function result 2-111

- Listing 2-1** Using the `WaitNextEvent` function 2-23
- Listing 2-2** An event loop 2-24
- Listing 2-3** Processing events 2-26
- Listing 2-4** The `Rez` input for a sample 'SIZE' resource 2-31
- Listing 2-5** Handling mouse-down events 2-34
- Listing 2-6** Handling key-down and auto-key events 2-44
- Listing 2-7** Handling key-down events 2-44
- Listing 2-8** Scanning for a Command-period event 2-46

Listing 2-9	Responding to update events	2-50
Listing 2-10	Responding to activate events	2-53
Listing 2-11	Responding to disk-inserted events	2-56
Listing 2-12	Handling null events	2-57
Listing 2-13	Responding to operating-system events	2-59
Listing 2-14	Responding to suspend and resume events	2-61
Listing 2-15	Changing the cursor	2-65
Listing 2-16	Accepting a high-level event	2-70
Listing 2-17	Posting a high-level event by application signature	2-74
Listing 2-18	Using the <code>PPCBrowser</code> function to post a high-level event	2-76
Listing 2-19	A Rez template for a 'SIZE' resource	2-116

Chapter 3

Menu Manager 3-1

Figure 3-1	A pull-down menu, a submenu, and a pop-up menu	3-6
Figure 3-2	The SurfWriter application's menu bar with the Edit menu displayed	3-7
Figure 3-3	The menu bar of the SurfWriter application	3-10
Figure 3-4	The SurfWriter application's menu bar localized for another script system	3-10
Figure 3-5	Two menus with various characteristics	3-13
Figure 3-6	Menu items in a mutually exclusive group	3-15
Figure 3-7	Menu items in an accumulating group	3-15
Figure 3-8	Use of a checkmark and dash in an accumulating group	3-15
Figure 3-9	The Apple menu for the SurfWriter application	3-21
Figure 3-10	Choosing the About command of the SurfWriter application	3-22
Figure 3-11	The standard File menu for an application	3-22
Figure 3-12	The standard Edit menu for an application	3-24
Figure 3-13	A typical Font menu	3-26
Figure 3-14	A Font menu showing a selection containing more than one font	3-27
Figure 3-15	A typical Size menu	3-28
Figure 3-16	A dialog box to select a new point size for a font	3-28
Figure 3-17	Entering a new point size for a font	3-29
Figure 3-18	The Other command with a font size added to it	3-29
Figure 3-19	The Help menu of the SurfWriter application	3-30
Figure 3-20	Default help balloons for the Apple menu and Application menu	3-31
Figure 3-21	Help balloons for different states of the Cut command	3-31
Figure 3-22	Accessing the Keyboard menu from an application	3-32
Figure 3-23	SurfWriter's Application menu	3-33
Figure 3-24	A pop-up menu	3-34
Figure 3-25	A pop-up menu in its closed and open states	3-34
Figure 3-26	Making a selection from a pop-up menu	3-35
Figure 3-27	Choosing one attribute from a list of many	3-36
Figure 3-28	A dialog box with checkboxes and pop-up menus	3-37
Figure 3-29	A type-in pop-up menu in its closed and open states	3-37
Figure 3-30	A type-in pop-up menu with a user's choice added	3-38
Figure 3-31	A hierarchical menu item and its submenu	3-39
Figure 3-32	A menu item with a submenu	3-53

Figure 3-33	A pop-up menu in a dialog box	3-56
Figure 3-34	Icons in menu items	3-63
Figure 3-35	A Size menu with user-specified size added	3-82
Figure 3-36	Menu access from a modal dialog box	3-85
Figure 3-37	Structure of a compiled menu ('MENU') resource	3-152
Figure 3-38	The variable-length data that describes menu items as defined by the standard menu definition procedure	3-153
Figure 3-39	Structure of a compiled menu bar ('MBAR') resource	3-155
Figure 3-40	Structure of a compiled menu color information table ('mctb') resource	3-156
Figure 3-41	Structure of a menu color entry in an 'mctb' resource	3-157
Table 3-1	Reserved keyboard equivalents for all systems	3-18
Table 3-2	Reserved keyboard equivalents for worldwide systems	3-19
Table 3-3	Other common keyboard equivalents	3-19
Table 3-4	Actions for standard File menu commands	3-23
Table 3-5	Actions for standard Edit menu commands	3-25
Table 3-6	Specifying submenus, script codes, reduced icons, small icons, and color icons of a menu item in a menu resource	3-46
Table 3-7	Color information for menu entries	3-100
Table 3-8	Mapping between new and previous names of Menu Manager routines	3-102
Listing 3-1	Rez input for a 'MENU' resource for the Apple menu	3-43
Listing 3-2	Rez input for a 'MENU' resource for an Edit menu	3-48
Listing 3-3	Rez input for a 'MENU' resource for a File menu	3-49
Listing 3-4	Rez input for an 'MBAR' resource	3-49
Listing 3-5	Setting up an application's menus and menu bar	3-50
Listing 3-6	Saving and restoring menu color information	3-52
Listing 3-7	Rez input for a description of a hierarchical menu with a submenu	3-54
Listing 3-8	Creating a hierarchical menu	3-55
Listing 3-9	Changing the text of a menu item	3-59
Listing 3-10	Setting the font style of menu items	3-60
Listing 3-11	Adding marks to and removing marks from menu items	3-61
Listing 3-12	Specifying icons for menu items	3-63
Listing 3-13	Rez input for text of menu items	3-66
Listing 3-14	Adding an item to the Help menu	3-68
Listing 3-15	Adding menu items to the Apple menu	3-69
Listing 3-16	Adding font names to a menu	3-70
Listing 3-17	Determining whether a mouse-down event occurred	3-72
Listing 3-18	Determining when the cursor is in the menu bar	3-72
Listing 3-19	Adjusting an application's menus	3-74
Listing 3-20	Adjusting the File menu for a document window	3-74
Listing 3-21	Adjusting the Edit menu for a document window	3-75
Listing 3-22	Determining when a key is pressed	3-77
Listing 3-23	Checking a key-down event for a keyboard equivalent	3-78
Listing 3-24	Responding to the user's choice of a menu command	3-79
Listing 3-25	Responding to the user's choice of an item from the Apple menu	3-80
Listing 3-26	Responding to the user's choice of a command from the Help menu	3-81

Listing 3-27	Handling the Size menu	3-83
Listing 3-28	A sample menu definition procedure	3-89
Listing 3-29	Calculating the size of a menu	3-90
Listing 3-30	Drawing menu items	3-91
Listing 3-31	Choosing menu items	3-93

Chapter 4

Window Manager 4-1

Figure 4-1	Multiple windows	4-4
Figure 4-2	A document window	4-5
Figure 4-3	Active and inactive document windows	4-7
Figure 4-4	A window of type <code>zoomDocProc</code>	4-8
Figure 4-5	A window of type <code>zoomDocProc</code> , with size box and inactive scroll bars	4-9
Figure 4-6	Window types for alert boxes and fixed-position modal dialog boxes	4-9
Figure 4-7	A window of type <code>movableDBoxProc</code>	4-10
Figure 4-8	A window of type <code>noGrowDocProc</code>	4-10
Figure 4-9	Seldom-used window types	4-11
Figure 4-10	Window frame, content region, and structure region	4-12
Figure 4-11	Scroll bars	4-14
Figure 4-12	Controls in a dialog box	4-15
Figure 4-13	The QuickDraw global coordinate plane	4-17
Figure 4-14	A window's local and global coordinate systems	4-19
Figure 4-15	Document window positions on a single screen	4-31
Figure 4-16	"Filling in" an empty document window position	4-31
Figure 4-17	Document window positions on multiple screens	4-33
Figure 4-18	Moving one window and adding to another window's update region	4-40
Figure 4-19	The close box with and without highlighting	4-46
Figure 4-20	The zoom box with and without highlighting	4-47
Figure 4-21	The effects of <code>BeginUpdate</code> and <code>EndUpdate</code> on the visible region and update region	4-49
Figure 4-22	The cumulative effects of <code>HideWindow</code> , <code>ShowWindow</code> , and <code>SelectWindow</code>	4-63
Figure 4-23	Limiting rectangle used by <code>DragGrayRgn</code>	4-98
Figure 4-24	Structure of a compiled window (' <code>WIND</code> ') resource	4-124
Figure 4-25	Structure of a compiled window color table (' <code>wctb</code> ') resource	4-128
Listing 4-1	Determining the window type	4-25
Listing 4-2	Rez input for a window (' <code>WIND</code> ') resource for a document window	4-26
Listing 4-3	Creating a new window	4-28
Listing 4-4	Application-defined data structure for storing a window's state data	4-34
Listing 4-5	Saving a document window's position	4-34
Listing 4-6	Positioning the window when the user opens a saved document	4-36
Listing 4-7	Opening a saved document	4-37
Listing 4-8	Drawing a window	4-39

Listing 4-9	Handling mouse-down events	4-44
Listing 4-10	Handling update events	4-50
Listing 4-11	Handling activate events	4-51
Listing 4-12	Zooming a window	4-55
Listing 4-13	Resizing a window	4-58
Listing 4-14	Adjusting scroll bars and content region when resizing a window	4-59
Listing 4-15	Converting a window region to local coordinates	4-60
Listing 4-16	Handling a close command	4-60
Listing 4-17	Closing a document	4-61
Listing 4-18	Showing a hidden dialog box	4-64

Chapter 5

Control Manager 5-1

Figure 5-1	Standard controls provided by the Control Manager	5-4
Figure 5-2	A default button	5-5
Figure 5-3	A selected checkbox	5-5
Figure 5-4	A vertical scroll bar	5-7
Figure 5-5	Using the scroll box and scroll arrows	5-8
Figure 5-6	Spatial relations between a document and a window, and their representation by a scroll bar	5-10
Figure 5-7	Custom slider controls	5-11
Figure 5-8	Visual feedback for user selection of active controls	5-12
Figure 5-9	Inactive controls	5-13
Figure 5-10	A button in a simple window	5-17
Figure 5-11	Radio buttons in a simple window	5-20
Figure 5-12	How a scroll bar should overlap the window frame	5-22
Figure 5-13	A pop-up menu	5-26
Figure 5-14	Dimensions of a sample pop-up menu	5-26
Figure 5-15	Three controls in a window	5-34
Figure 5-16	Moving a document relative to its window	5-46
Figure 5-17	Updating the contents of a scrolled window	5-49
Figure 5-18	Restoring the window origin to (0,0)	5-50
Figure 5-19	Scrolling to the end of a document	5-51
Figure 5-20	Updating a window's contents and returning the window origin to (0,0)	5-51
Figure 5-21	Moving and resizing scroll bars	5-66
Figure 5-22	A vertical scroll bar before the application moves it within a resized window	5-69
Figure 5-23	A vertical scroll bar after the application moves its upper-left point	5-69
Figure 5-24	A custom control	5-71
Figure 5-25	Structure of a compiled control ('CNTL') resource	5-118
Figure 5-26	Structure of a compiled control color table ('cctb') resource	5-122
Table 5-1	Mapping between new and previous names of Control Manager routines	5-80
Listing 5-1	Creating a button for a window	5-17
Listing 5-2	Rez input for a control resource	5-18

Listing 5-3	Rez input for the control resources of radio buttons	5-21
Listing 5-4	Rez input for resources for a window and its scroll bars	5-23
Listing 5-5	Creating a document window with scroll bars	5-24
Listing 5-6	Rez input for the control resource of a pop-up menu	5-26
Listing 5-7	Responding to an update event for a window	5-29
Listing 5-8	Redrawing the controls in the update region	5-30
Listing 5-9	Detecting mouse-down events in a window	5-32
Listing 5-10	Detecting mouse-down events in a pop-up menu and a button	5-33
Listing 5-11	Using the <code>TrackControl</code> function with a button	5-36
Listing 5-12	Using <code>TrackControl</code> with a pop-up menu	5-37
Listing 5-13	Responding to a click in a checkbox	5-38
Listing 5-14	Adjusting scroll bar settings and locations	5-39
Listing 5-15	Assigning settings to scroll bars	5-40
Listing 5-16	Adjusting the maximum and current settings for a scroll bar	5-41
Listing 5-17	Using <code>ScrollRect</code> to scroll the bits displayed in the window	5-47
Listing 5-18	Responding to mouse events in a scroll bar	5-53
Listing 5-19	Action procedures for scrolling through a text document	5-59
Listing 5-20	Moving the scroll box from the action procedures	5-61
Listing 5-21	An application-defined update routine	5-62
Listing 5-22	Redrawing a window containing graphics objects	5-63
Listing 5-23	Redrawing a window after scrolling a <code>TextEdit</code> edit record	5-65
Listing 5-24	Changing the size and location of a window's scroll bars	5-67

Chapter 6

Dialog Manager 6-1

Figure 6-1	An alert box used by the Finder	6-6
Figure 6-2	A typical dialog box	6-7
Figure 6-3	A note alert	6-8
Figure 6-4	A caution alert	6-9
Figure 6-5	A stop alert	6-9
Figure 6-6	A modal dialog box	6-10
Figure 6-7	A movable modal dialog box	6-11
Figure 6-8	A modeless dialog box	6-12
Figure 6-9	Typical items in a dialog box	6-13
Figure 6-10	An alert box to save changes to a document	6-19
Figure 6-11	An alert box displayed only during the third and fourth alert stages	6-21
Figure 6-12	A simple modal dialog box	6-24
Figure 6-13	Relationship of various resources to an alert box	6-27
Figure 6-14	A safe default button in an alert box	6-31
Figure 6-15	The consistent spacing of buttons and text in an alert box	6-33
Figure 6-16	Incorrectly and correctly sized display rectangles for alternate script systems	6-35
Figure 6-17	Inactive controls and disabled items	6-37
Figure 6-18	A dialog box with OK and Cancel buttons	6-38
Figure 6-19	A movable modal dialog box with a Stop button	6-39
Figure 6-20	An alert box with a Revert button	6-39
Figure 6-21	An obscure and useless alert message	6-41

Figure 6-22	A less obscure alert message	6-41
Figure 6-23	A clear and helpful alert message	6-41
Figure 6-24	A pop-up menu in a dialog box	6-42
Figure 6-25	A selected scrolling list	6-45
Figure 6-26	An alert box that displays a document name	6-46
Figure 6-27	Two editable text items in a modeless dialog box	6-48
Figure 6-28	An existing dialog box and items to append	6-51
Figure 6-29	The dialog box after items are overlaid	6-52
Figure 6-30	The dialog box after items are appended to the right	6-52
Figure 6-31	The dialog box after items are appended to the bottom	6-53
Figure 6-32	A dialog box with an item appended relative to an existing item	6-53
Figure 6-33	An alert box in front of a document window	6-63
Figure 6-34	An alert box on the main screen	6-63
Figure 6-35	An alert box in the alert position of the document window screen	6-64
Figure 6-36	An alert box displayed only after the third alert stage	6-65
Figure 6-37	A modeless dialog box for changing text in a document	6-66
Figure 6-38	Menu access when displaying a modal dialog box	6-69
Figure 6-39	Three buttons for which <code>CautionAlert</code> reports events	6-81
Figure 6-40	Four items for which <code>ModalDialog</code> reports events	6-83
Figure 6-41	A modeless dialog box for which <code>DialogSelect</code> reports events	6-91
Figure 6-42	Structure of a compiled dialog ('DLOG') resource	6-148
Figure 6-43	Structure of a compiled alert ('ALRT') resource	6-150
Figure 6-44	Structure of a compiled item list ('DITL') resource	6-152
Figure 6-45	Structure of compiled button, checkbox, radio button, static text, and editable text items	6-153
Figure 6-46	Structure of compiled control, icon, and picture items	6-154
Figure 6-47	Structure of a compiled application-defined item	6-155
Figure 6-48	Structure of compiled help items	6-155
Figure 6-49	Structure of a compiled item color table resource	6-159
Figure 6-50	Structure of a compiled control color table	6-161
Figure 6-51	Structure of a compiled text style table	6-162
Table 6-1	Mapping between new and previous names of Dialog Manager routines	6-102
Listing 6-1	Rez input for an alert resource	6-19
Listing 6-2	Specifying different alert responses according to alert stage	6-21
Listing 6-3	Creating your own sound procedure for alerts	6-22
Listing 6-4	Rez input for a dialog resource	6-24
Listing 6-5	Rez input for providing an alert box with items	6-27
Listing 6-6	Rez input for consistent spacing of display rectangles	6-34
Listing 6-7	Rez input for a dialog resource and an item list resource for a dialog box that includes a pop-up menu	6-43
Listing 6-8	Rez input for a control resource and a menu resource for a pop-up menu	6-43
Listing 6-9	Using the <code>ParamText</code> procedure to substitute text strings	6-47

Listing 6-10	Specifying where <code>ParamText</code> should substitute text in an alert box message	6-48
Listing 6-11	Specifying editable text items in an item list	6-49
Listing 6-12	Getting the text entered by the user in an editable text item	6-49
Listing 6-13	Appending an item to an existing dialog box	6-54
Listing 6-14	Rez input for a dialog box and the item appended to it	6-55
Listing 6-15	Rez input for an application-defined item in an item list	6-57
Listing 6-16	Installing the draw procedure for an application-defined item	6-58
Listing 6-17	Creating a draw procedure that draws a bold outline around the default button	6-59
Listing 6-18	Deactivating the front window before displaying an alert box	6-65
Listing 6-19	Using <code>GetAlertStage</code> to determine when to deactivate the front window	6-66
Listing 6-20	Ensuring that the modeless dialog box isn't already open before creating it	6-67
Listing 6-21	Adjusting menus for various windows	6-70
Listing 6-22	Disabling menus for a modal dialog box with editable text items	6-70
Listing 6-23	Adjusting the Edit menu for a modal dialog box	6-72
Listing 6-24	Rez input for a dialog color table resource using the system's default colors	6-75
Listing 6-25	Using <code>DialogSelect</code> during null events	6-79
Listing 6-26	Responding to events in a modal dialog box	6-83
Listing 6-27	A typical event filter function for alert and modal dialog boxes	6-88
Listing 6-28	Handling mouse-down events for all windows	6-91
Listing 6-29	Using the <code>DialogSelect</code> function for responding to mouse-down events	6-92
Listing 6-30	Hiding a modeless dialog box in response to a Close command	6-94
Listing 6-31	Checking for key-down events involving the Command key	6-95
Listing 6-32	Checking for key-down events in a modeless dialog box	6-95
Listing 6-33	Responding to key-down events in a modeless dialog box	6-96
Listing 6-34	Activating a modeless dialog box	6-98
Listing 6-35	Updating a modeless dialog box	6-99

Chapter 7

	Finder Interface	7-1
Figure 7-1	Application and document icons in a window on the desktop	7-4
Figure 7-2	A customized help balloon for an application icon	7-5
Figure 7-3	A Finder message identifying a missing application	7-5
Figure 7-4	Large black-and-white application icons for a company's product line	7-12
Figure 7-5	Default large black-and-white icons	7-12
Figure 7-6	A black-and-white icon and its mask for an application	7-13
Figure 7-7	The ResEdit view of an icon	7-14
Figure 7-8	Linking icon list resources and file reference resources in a bundle resource	7-23
Figure 7-9	The default application-unavailable alert box	7-27

Figure 7-10	The application-unavailable alert box specifying an application's name	7-29
Figure 7-11	The application-unavailable alert box with a customized message	7-30
Figure 7-12	The application-unavailable alert box for 'TEXT' and 'PICT' documents	7-30
Figure 7-13	The version data in the information window	7-33
Figure 7-14	Default and customized help balloons for application icons	7-38
Figure 7-15	The System Folder and related folders	7-42
Figure 7-16	Structure of a signature resource compiled as a string ('STR') resource	7-57
Figure 7-17	Structure of a compiled icon list ('ICON#') resource	7-58
Figure 7-18	Structure of a compiled small icon list ('ics#') resource	7-59
Figure 7-19	Structure of a compiled large 4-bit color icon ('icl4') resource	7-60
Figure 7-20	Structure of a compiled small 4-bit color icon ('ics4') resource	7-61
Figure 7-21	Structure of a compiled large 8-bit color icon ('icl8') resource	7-62
Figure 7-22	Structure of a compiled small 8-bit color icon ('ics8') resource	7-63
Figure 7-23	Structure of a compiled icon ('ICON') resource	7-64
Figure 7-24	Structure of a compiled file reference ('FREF') resource	7-65
Figure 7-25	Structure of a compiled bundle ('BNDL') resource	7-66
Figure 7-26	Mapping local IDs to icon list resource IDs in a bundle resource	7-67
Figure 7-27	Structure of superfluous local ID mapping for file reference resources in a bundle resource	7-67
Figure 7-28	Structure of a compiled missing-application name string resource	7-68
Figure 7-29	Structure of a compiled application-missing message string resource	7-69
Figure 7-30	Format of a compiled version ('vers') resource	7-70
Listing 7-1	Rez input for a signature resource	7-8
Listing 7-2	Rez input for an icon list resource	7-14
Listing 7-3	Rez input for file reference resources	7-19
Listing 7-4	Rez input for a bundle resource	7-21
Listing 7-5	Rez input for a missing-application name string resource	7-28
Listing 7-6	Storing a missing-application name string resource in the resource fork of a document	7-28
Listing 7-7	Copying the missing-application name string resource into the resource fork of a document	7-29
Listing 7-8	Rez input for an application-missing message string resource	7-30
Listing 7-9	Rez input for a pair of version resources	7-32
Listing 7-10	Rez input for a size resource	7-35
Listing 7-11	Determining whether a document is a stationery pad	7-36
Listing 7-12	Rez input for a help balloon resource for an application icon	7-39
Listing 7-13	Using the <code>ResolveAliasFile</code> function to open a file	7-41

About This Book

This book, *Inside Macintosh: Macintosh Toolbox Essentials*, describes the essential elements of a Macintosh application and the system software routines that you can use to implement them.

If you are new to programming on the Macintosh computer, you should also read *Inside Macintosh: Overview* for an introduction to general concepts of Macintosh programming and *Macintosh Human Interface Guidelines* for a complete discussion of user interface guidelines and principles that every Macintosh application should follow.

This book describes events, windows, menus, controls, alert boxes, and dialog boxes. It also discusses how your application interacts with the Finder.

Macintosh applications respond to user actions and to other hardware- and software-related events. To design your application so that it can respond to events (such as keyboard input, mouse input, changes in the appearance of windows on the screen, and changes in your application's processing status), see the chapter "Event Manager" in this book.

To create menus and set up your application's menu bar, see the chapter "Menu Manager." This chapter describes how to define the items in your menus, how to enable and disable menus, how to allow the user to choose a menu item, and how to respond once the user chooses a menu item.

To create windows in which the user can view or edit information, see the chapter "Window Manager." This chapter describes the basic types of windows and discusses how your application can work together with the Window Manager to support the standard user interface conventions associated with manipulating a window, such as moving a window, zooming a window, and resizing a window.

To create controls in your application's windows—such as scroll bars—or to create controls in dialog boxes—such as buttons or checkboxes—see the chapter "Control Manager."

To create dialog boxes or alert boxes—windows that your application uses to communicate with or solicit information from the user—see the chapter "Dialog Manager."

To create icons for your applications and the documents it creates, see the chapter "Finder Interface." This chapter also introduces file types and creators and describes the various kinds of resources (icons, file references, and bundles) that the Finder needs to display your application and the documents it creates.

After implementing the basic elements of a Macintosh application as described in this book, you can add additional features, such as help balloons

PREFACE

and support for copy and paste, as described in *Inside Macintosh: More Macintosh Toolbox*. You can also find detailed information about the Resource Manager in *Inside Macintosh: More Macintosh Toolbox*.

Once you understand how to create menus, windows, and dialog boxes, you can save information that the user enters in a window by writing the data to a file. You can also open a previously saved file and read the information from the file into a window. You use the File Manager to open, read, write, and close files. See the chapter “Introduction to File Management” in *Inside Macintosh: Files* for information on how to read and write files.

For information about drawing into a window or other graphics port, see *Inside Macintosh: Imaging*.

For information on handling text in your application, see *Inside Macintosh: Text*.

For information on communicating with other applications, see *Inside Macintosh: Interapplication Communication*.

Format of a Typical Chapter

Almost all chapters in this book follow a standard structure. For example, the Event Manager chapter contains these sections:

- “Introduction to Events.” This section presents a general introduction to the types of events that your application can receive.
- “About the Event Manager.” This section provides an overview of the features provided by the Event Manager.
- “Using the Event Manager.” This section describes the tasks you can accomplish using the Event Manager. It describes how to use the most common routines, gives related user interface information, provides code samples, and supplies additional information.
- “Event Manager Reference.” This section provides a complete reference to the Event Manager by describing the data structures, routines, and resources it uses. Each routine description also follows a standard format, which presents the routine declaration followed by 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 Event Manager.” This section provides the Pascal and C interfaces for the constants, data structures, routines, and result codes associated with the Event Manager. It also includes relevant assembly-language interface information.

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 the contents of registers, use special formats so that you can scan them quickly.

PREFACE

Special Fonts

All code listings, reserved words, and names of actual data structures, fields, constants, 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 2-7.) ♦

IMPORTANT

A note like this contains information that is essential for an understanding of the main text. (An example appears on page 5-27.) ▲

▲ 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 2-105.) ▲

Empty Strings

This book occasionally instructs you to provide an empty string in routine parameters and resources. How you specify an empty string depends on what language and development environment you are using. In Rez input files and in C code, for example, you specify an empty string by using two double quotation marks (""), and in Pascal you specify an empty string by using two single quotation marks ('').

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

PREFACE

In the “Assembly-Language Summary” section at the end of each chapter, *Inside Macintosh* presents information about the fields of data structures in this format:

0	what	word	event code
2	message	long	event message
6	when	long	ticks since startup

The left column indicates the byte offset of the field from the beginning of the data structure. The second column shows the field name as defined in the MPW Pascal interface files; the third column indicates the size 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 of the data structure in the reference section of the chapter.

The 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. When showing system software routines, this book uses the Pascal interface available with the Macintosh Programmer’s Workshop (MPW).

All code listings in this book are shown in Pascal (except for listings that describe resources, which are shown in Rez-input format). They show methods of using various routines and illustrate techniques for accomplishing particular tasks. All code listings have been compiled and, in many cases, tested. However, Apple Computer, Inc., does not intend for you to use these code samples in your application. You can find the location of code listings in the list of figures, tables, and listings. If you know the name of a particular routine (such as `DoEvent` or `MyAdjustMenus`) shown in a code listing, you can find the page on which the routine occurs by looking under the entry “sample routines” in the index of this book.

In order to make the code listings in this book more readable, they show only limited error handling. You need to develop your own techniques for handling errors.

This book occasionally illustrates concepts by reference to a sample application called *SurfWriter*; this is not an actual product of Apple Computer, Inc.

APDA is Apple’s worldwide source for over three hundred development tools, technical resources, training products, and information for anyone interested in developing applications on Apple platforms. Customers receive the quarterly *APDA Tools Catalog* featuring all current versions of Apple and the most popular third-party development tools. Ordering is easy; there are no membership fees, and application forms are not required for most products. APDA offers convenient payment and shipping options including site licensing.

PREFACE

To order products or to request a complimentary copy of the *APDA Tools Catalog*, contact:

APDA
Apple Computer, Inc.
P.O. Box 319
Buffalo, NY 14207-0319

Telephone: 800-282-2732 (United States)
800-637-0029 (Canada)
716-871-6555 (elsewhere in the world)

Fax: 716-871-6511

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

