

INSIDE MACINTOSH

More Macintosh Toolbox



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.
© 1993, 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, AppleLink, AppleShare, AppleTalk, EtherTalk, ImageWriter, LaserWriter, LocalTalk, Macintosh, MPW, StyleWriter, and TokenTalk are trademarks of Apple Computer, Inc., registered in the United States and other countries.

Apple Desktop Bus, Balloon Help, BalloonWriter, Chicago, Finder, Geneva, KanjiTalk, QuickDraw, QuickTime, ResEdit, System 7, and System 7.0 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.

America Online is a service mark of Quantum Computer Services, Inc.

CompuServe is a registered service mark of CompuServe, Inc.

FrameMaker is a registered trademark of Frame Technology Corporation.

Helvetica and Palatino are registered trademarks of Linotype Company.

Internet is a trademark of Digital Equipment Corporation.

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 0-201-63299-3
1 2 3 4 5 6 7 8 9-CRW-9796959493
First Printing, October 1993

Library of Congress Cataloging-in-Publication Data

Inside Macintosh : more Macintosh toolbox / [Apple Computer, Inc.]
p. cm.
Includes index.
ISBN 0-201-63299-3
1. Macintosh (Computer)—Programming. 2. Macintosh Toolbox.
I. Apple Computer, Inc.
QA76.8.M315613 1993
005.265—dc20

93-33260
CIP

Contents

Figures, Tables, and Listings xvii

Preface

About This Book xxv

Format of a Typical Chapter xxvi
Conventions Used in This Book xxvii
 Special Fonts xxvii
 Types of Notes xxvii
 Empty Strings xxviii
 Assembly-Language Information xxviii
The Development Environment xxviii

Chapter 1

Resource Manager 1-1

Introduction to Resources 1-3
 The Data Fork and the Resource Fork 1-4
 Resource Types and Resource IDs 1-6
 The Resource Map 1-8
 Search Path for Resources 1-10
About the Resource Manager 1-12
Using the Resource Manager 1-13
 Creating a Resource 1-15
 Getting a Resource 1-18
 Releasing and Detaching Resources 1-22
 Opening a Resource Fork 1-24
 Opening an Application's Resource Fork 1-24
 Creating and Opening a Resource Fork 1-25
 Specifying the Current Resource File 1-28
 Reading and Manipulating Resources 1-30
 Writing Resources 1-36
 Working With Partial Resources 1-40
Resource Manager Reference 1-42
 Data Structure, Resource Types, and Resource IDs 1-42
 The Resource Type 1-42
 Resource IDs 1-46
 Resource IDs of Owned Resources 1-47
 Resource Names 1-49
 Resource Manager Routines 1-49
 Initializing the Resource Manager 1-50
 Checking for Errors 1-51
 Creating an Empty Resource Fork 1-53

Opening Resource Forks	1-58
Getting and Setting the Current Resource File	1-68
Reading Resources Into Memory	1-71
Getting and Setting Resource Information	1-81
Modifying Resources	1-87
Writing to Resource Forks	1-92
Getting a Unique Resource ID	1-95
Counting and Listing Resource Types	1-97
Getting Resource Sizes	1-104
Disposing of Resources	1-106
Closing Resource Forks	1-110
Reading and Writing Partial Resources	1-111
Getting and Setting Resource Fork Attributes	1-116
Accessing Resource Entries in a Resource Map	1-119
Resource File Format	1-121
Resources in the System File	1-126
User Information Resources	1-127
Packages	1-128
Function Key Resources	1-129
Standard Icons	1-129
ROM Resources	1-134
Inserting the ROM Resource Map	1-134
Overriding ROM Resources	1-135
Summary of the Resource Manager	1-137
Pascal Summary	1-137
Constants	1-137
Data Type	1-139
Routines	1-139
C Summary	1-142
Constants	1-142
Data Type	1-143
Routines	1-144
Assembly-Language Summary	1-147
Trap Macros	1-147
Global Variables	1-147
Result Codes	1-148

Chapter 2

Scrap Manager 2-1

Introduction to the Scrap Manager	2-4
The Clipboard	2-10
Intelligent Cut and Paste	2-10
About the Scrap Manager	2-12
Location of the Scrap	2-12
Using the Scrap Manager	2-14
Getting Information About the Scrap	2-15

Putting Data in the Scrap	2-15
Handling the Cut Command	2-15
Handling the Copy Command	2-19
Handling Suspend Events	2-19
Getting Data From the Scrap	2-20
Handling the Paste Command	2-20
Handling Resume Events	2-25
Converting Data Between a Private Scrap and the Scrap	2-26
Converting Data Between the TextEdit Scrap and the Scrap	2-28
Handling Editing Operations in Dialog Boxes	2-31
Scrap Manager Reference	2-31
Data Structures	2-32
The Scrap Information Record	2-32
The Scrap Format Types	2-33
Routines	2-34
Getting Information About the Scrap	2-34
Writing Information to the Scrap	2-35
Reading Information From the Scrap	2-38
Transferring Data Between the Scrap in Memory and the Scrap on Disk	2-40
Summary of the Scrap Manager	2-42
Pascal Summary	2-42
Constants	2-42
Data Types	2-42
Routines	2-42
C Summary	2-43
Data Types	2-43
Routines	2-44
Assembly-Language Summary	2-45
Data Structures	2-45
Result Codes	2-45

Chapter 3

Help Manager 3-1

About the Help Manager	3-6
How the Help Manager Displays Balloons	3-8
Default Help Balloons for Menus, Windows, and Icons	3-13
About BalloonWriter	3-17
Using the Help Manager	3-18
Providing Text or Pictures for Help Balloons	3-18
Defining Help Messages	3-19
Using Clear, Concise Phrases	3-20
Using Active Constructions	3-22
Using Parallel Structure	3-22
Offering Hints	3-22
Using Consistent Terminology	3-23

Defining the Help Balloon Position	3-23
Specifying the Format for Help Messages	3-23
Specifying Options in Help Resources	3-25
Providing Help Balloons for Menus	3-27
Specifying Header Information for the 'hmnu' Resource	3-32
Specifying Help for Menu Items Missing From the Resource	3-33
Specifying Help for Menu Titles and for Items Dimmed by System Software	3-36
Specifying Help for Menu Items	3-39
Specifying Help for a Changing Menu Item	3-43
Specifying Resources by Item Name	3-45
Providing Help Balloons for Menus You Disable for Dialog Boxes	3-47
Providing Help Balloons for Items in Dialog Boxes and Alert Boxes	3-51
Specifying Header Information for the 'hdlg' Resource	3-54
Specifying Missing-Item Information	3-54
Specifying Help for Items in an Alert or Dialog Box	3-56
Adding a Help Item to an Item List Resource	3-62
Using a Help Item Versus Using an 'hwin' Resource	3-63
Providing Help Balloons for Window Content	3-63
Providing Help Balloons for Static Windows	3-65
Specifying Header Information for the 'hrct' Resource	3-67
Specifying Help for Rectangles in Windows	3-67
Associating Help Resources With Static Windows	3-68
Specifying Header Information for the 'hwin' Resource	3-69
Specifying 'hdlg' or 'hrct' Resources in the 'hwin' Resource	3-69
Providing Help Balloons for Dynamic Windows	3-74
Overriding Help Balloons for Non-Document Icons	3-84
Specifying Header Information for the 'hldr' Resource	3-85
Specifying Help for an Icon	3-85
Overriding Other Default Help Balloons	3-87
Specifying Header Information for the 'hovr' Resource	3-88
Overriding Default Help	3-88
Adding Menu Items to the Help Menu	3-90
Writing Your Own Balloon Definition Function	3-93
Help Manager Reference	3-95
Data Structures	3-95
The Help Message Record	3-95
The Help Manager String List Record	3-97
Help Manager Routines	3-97
Determining Balloon Help Status	3-98
Displaying and Removing Help Balloons	3-99
Enabling and Disabling Balloon Help Assistance	3-107
Adding Items to the Help Menu	3-108
Getting and Setting the Font Name and Size	3-110
Setting and Getting Information for Help Resources	3-114
Determining the Size of a Help Balloon	3-119
Getting the Message of a Help Balloon	3-122

Application-Defined Routines	3-128
Resources	3-132
The Menu Help Resource	3-132
The Dialog-Item Help Resource	3-140
The Rectangle Help Resource	3-148
The Window Help Resource	3-154
The Finder Icon Help Resource	3-156
The Default Help Override Resource	3-160
Summary of the Help Manager	3-166
Pascal Summary	3-166
Constants	3-166
Data Types	3-168
Help Manager Routines	3-169
Application-Defined Routines	3-170
C Summary	3-170
Constants	3-170
Data Types	3-173
Help Manager Routines	3-173
Application-Defined Routines	3-175
Assembly-Language Summary	3-176
Data Structures	3-176
Trap Macros	3-176
Result Codes	3-177

Chapter 4

List Manager 4-1

Introduction to Lists	4-4
Appearance of Lists	4-4
Selection of List Items	4-9
Keyboard Navigation of Lists	4-15
Movement of a Selection With Arrow Keys	4-15
Extension of a Selection With Arrow Keys	4-16
Type Selection in a Text-Only List	4-20
Multiple Lists in a Window	4-20
About the List Manager	4-22
Using the List Manager	4-26
Creating a List	4-27
Adding Rows and Columns to a List	4-30
Responding to Events Affecting a List	4-32
Working With List Selections	4-34
Customizing Cell Highlighting	4-38
Manipulating List Cells	4-40
Searching a List for a Particular Item	4-43
Supporting Keyboard Navigation of Lists	4-45
Supporting Type Selection of List Items	4-45
Supporting Arrow-Key Navigation of Lists	4-48

Supporting the Anchor Algorithm for Extending Lists With Arrow Keys	4-52	
Outlining the Current List	4-53	
Writing Your Own List Definition Procedure	4-58	
Responding to the Initialization Message	4-60	
Responding to the Draw Message	4-60	
Responding to the Highlighting Message	4-62	
Responding to the Close Message	4-62	
Using the Pictures List Definition Procedure	4-63	
List Manager Reference	4-65	
Data Structures	4-65	
The Cell Record	4-65	
The Data Handle	4-66	
The List Record	4-66	
List Manager Routines	4-70	
Creating and Disposing of Lists	4-70	
Adding and Deleting Columns and Rows To and From a List	4-73	
Determining or Changing the Selection	4-77	
Accessing and Manipulating Cell Data	4-79	
Responding to Events Affecting Lists	4-84	
Modifying a List's Appearance	4-87	
Searching a List for a Particular Item	4-90	
Changing the Size of Cells and Lists	4-91	
Getting Information About Cells	4-93	
Application-Defined Routines	4-96	
List Definition Procedures	4-96	
Match Functions	4-99	
Click-Loop Procedures	4-100	
Summary of the List Manager	4-102	
Pascal Summary	4-102	
Constants	4-102	
Data Types	4-102	
List Manager Routines	4-103	
Application-Defined Routines	4-105	
C Summary	4-106	
Constants	4-106	
Data Types	4-106	
List Manager Routines	4-107	
Application-Defined Routines	4-109	
Assembly-Language Summary	4-110	
Data Structures	4-110	
Trap Macros	4-111	

Introduction to the Icon Utilities	5-3
About the Icon Utilities	5-6
Using the Icon Utilities	5-7
Drawing Icons in an Icon Family	5-8
Drawing an Icon Directly From a Resource	5-10
Getting an Icon Suite and Drawing One of Its Icons	5-11
Drawing Specific Icons From an Icon Family	5-12
Manipulating Icons	5-13
Drawing Icons That Are Not Part of an Icon Family	5-13
Icon Utilities Reference	5-17
Data Structure	5-17
The Color Icon Record	5-17
Icon Utilities Routines	5-18
Drawing Icons From Resources	5-19
Getting Icons From Resources That Don't Belong to an Icon Family	5-28
Disposing of Icons	5-30
Creating an Icon Suite	5-30
Getting Icons From an Icon Suite	5-34
Drawing Icons From an Icon Suite	5-35
Performing Operations on Icons in an Icon Suite	5-38
Getting and Setting the Label for an Icon Suite	5-40
Getting Label Information	5-41
Disposing of Icon Suites	5-42
Converting an Icon Mask to a Region	5-43
Determining Whether a Point or Rectangle Is Within an Icon	5-46
Working With Icon Caches	5-53
Application-Defined Routines	5-57
Icon Action Functions	5-57
Icon Getter Functions	5-58
Summary of the Icon Utilities	5-60
Pascal Summary	5-60
Constants	5-60
Data Types	5-62
Icon Utilities Routines	5-62
Application-Defined Routines	5-65
C Summary	5-65
Constants	5-65
Data Types	5-67
Icon Utilities Routines	5-68
Application-Defined Routines	5-71
Assembly-Language Summary	5-71
Data Structure	5-71
Trap Macros	5-72
Result Codes	5-73

Introduction to Components	6-3	
About the Component Manager	6-4	
Using the Component Manager	6-6	
Opening Connections to Components	6-7	
Opening a Connection to a Default Component		6-7
Finding a Specific Component	6-8	
Opening a Connection to a Specific Component		6-9
Getting Information About a Component	6-10	
Using a Component	6-11	
Closing a Connection to a Component	6-12	
Creating Components	6-13	
The Structure of a Component	6-13	
Handling Requests for Service	6-18	
Responding to the Open Request	6-19	
Responding to the Close Request	6-21	
Responding to the Can Do Request	6-22	
Responding to the Version Request	6-22	
Responding to the Register Request	6-23	
Responding to the Unregister Request	6-24	
Responding to the Target Request	6-25	
Responding to Component-Specific Requests		6-26
Reporting an Error Code	6-28	
Defining a Component's Interfaces	6-28	
Managing Components	6-30	
Registering a Component	6-30	
Creating a Component Resource	6-32	
Establishing and Managing Connections		6-34
Component Manager Reference	6-37	
Data Structures for Applications	6-37	
The Component Description Record	6-37	
Component Identifiers and Component Instances		6-40
Routines for Applications	6-41	
Finding Components	6-42	
Opening and Closing Components	6-44	
Getting Information About Components		6-47
Retrieving Component Errors	6-51	
Data Structures for Components	6-52	
The Component Description Record	6-52	
The Component Parameters Record	6-54	
Routines for Components	6-56	
Registering Components	6-57	
Dispatching to Component Routines	6-63	
Managing Component Connections	6-65	
Setting Component Errors	6-69	
Working With Component Reference Constants		6-70

Accessing a Component's Resource File	6-71
Calling Other Components	6-73
Capturing Components	6-75
Targeting a Component Instance	6-77
Changing the Default Search Order	6-78
Application-Defined Routines	6-79
Resources	6-80
The Component Resource	6-80
Summary of the Component Manager	6-86
Pascal Summary	6-86
Constants	6-86
Data Types	6-87
Routines for Applications	6-89
Routines for Components	6-90
Application-Defined Routines	6-92
C Summary	6-92
Constants	6-92
Data Structures	6-93
Routines for Applications	6-95
Routines for Components	6-96
Application-Defined Routines	6-97
Assembly-Language Summary	6-98
Trap Macros	6-98
Result Codes	6-99

Chapter 7

Translation Manager 7-1

About the Translation Manager	7-4
Opening Documents From the Finder	7-5
Opening Documents Within an Application	7-8
Translating Documents on the Desktop	7-9
Sharing Data Between Applications	7-10
Using the Translation Manager	7-10
Checking for the Translation Manager	7-12
Declaring the File Types Your Application Can Open	7-13
Declaring Custom Kind Strings	7-14
Using File-Opening Dialog Boxes	7-15
Translating Files Explicitly	7-17
Writing a Translation Extension	7-18
Creating a Translation Extension	7-19
Dispatching to Translation Extension-Defined Routines	7-24
Creating a Translation List	7-27
Identifying Files	7-32
Translating Files	7-33
Writing Application Translation Extensions	7-35

Translation Manager Reference	7-36	
Translation Manager Routines	7-36	
Getting Translation Information	7-37	
Translating Files	7-42	
Resources	7-43	
The Open Resource	7-44	
The Kind Resource	7-45	
Translation Extension Reference	7-46	
Translation Extension Data Structures	7-46	
File Type Specifications	7-46	
File Translation Lists	7-48	
Scrap Type Specifications	7-49	
Scrap Translation Lists	7-49	
Translation Extension Routines	7-50	
Managing Translation Progress Dialog Boxes	7-50	
Translation Extension-Defined Routines	7-54	
File Translation Extension Routines	7-54	
Scrap Translation Extension Routines	7-58	
Summary of the Translation Manager	7-63	
Pascal Summary	7-63	
Constants	7-63	
Data Types	7-63	
Translation Manager Routines	7-64	
C Summary	7-64	
Constants	7-64	
Data Types	7-65	
Translation Manager Routines	7-65	
Assembly-Language Summary	7-66	
Data Structures	7-66	
Trap Macros	7-66	
Result Codes	7-67	
Summary of Translation Extensions	7-68	
Pascal Summary	7-68	
Constants	7-68	
Data Types	7-68	
Translation Extension Routines	7-70	
Translation Extension-Defined Routines	7-70	
C Summary	7-71	
Constants	7-71	
Data Types	7-71	
Translation Extension Routines	7-73	
Translation Extension-Defined Routines	7-73	
Assembly-Language Summary	7-74	
Data Structures	7-74	
Trap Macros	7-75	
Result Codes	7-75	

About Control Panels	8-4
Control Panels	8-4
A Control Panel's Resources	8-6
The Finder's Interaction With Control Panels	8-7
Control Panels and System Extensions	8-8
About User Documentation for Control Panels	8-8
The Monitors Control Panel and Extensions to It	8-9
Creating Control Panel Files	8-12
Defining the User Interface for a Control Panel	8-12
Creating a Control Panel's Resources	8-14
Resource IDs for Control Panels	8-14
Defining the Control Panel Rectangles	8-15
Creating the Item List Resource	8-17
Defining the Icon for a Control Panel	8-20
Specifying the Machine Resource	8-20
Creating the File Reference, Bundle, and Signature Resources	8-21
Providing Additional Resources for a Control Panel	8-22
Specifying the Font of Text in a Control Panel	8-23
Creating a Font Information Resource	8-23
Defining Text in a Control Panel as User Items	8-24
Writing a Control Panel Function	8-25
Determining If a Control Panel Can Run on the Current System	8-29
Initializing the Control Panel Items and Allocating Storage	8-29
Responding to Activate Events	8-33
Responding to Keyboard Events	8-37
Responding to Mouse Events	8-39
Responding to Update Events	8-43
Handling Text Defined as User Items	8-43
Responding to Null Events	8-45
Responding to the User Closing the Control Panel	8-45
Handling Edit Menu Commands	8-46
Handling Errors	8-47
Creating an Extension for the Monitors Control Panel	8-48
Designing the User Interface for a Monitors Extension	8-49
Creating the Required Resources for a Monitors Extension	8-51
Creating a Card Resource for a Monitors Extension	8-51
Defining a Rectangle for a Monitors Extension	8-52
Creating an Item List Resource for a Monitors Extension	8-54
Creating the Monitor Code Resource	8-56
Supplying Optional Resources for a Monitors Extension	8-56
Specifying an Icon for the Options Dialog Box	8-57
Specifying Version Information	8-58
Providing an Alternative Name for a Video Card	8-58
Supplying Gamma Table Resources	8-59
Creating File Reference, Bundle, and Signature Resources	8-59

Including a System Extension Resource	8-61
Writing a Monitors Extension Function	8-61
Handling the Startup Message	8-66
Performing Initialization	8-68
Responding to a Click in the OK Button	8-70
Responding to a Cancel Request	8-71
Handling Mouse Events for a Monitors Extension	8-71
Handling Keyboard Events	8-73
Including Another Control Panel Definition in a Monitors Extension File	8-73
Control Panels Reference	8-74
Application-Defined Routines	8-74
Control Device Functions	8-74
Monitors Extension Functions	8-78
Resources	8-82
The Machine Resource	8-84
The Rectangle Positions Resource	8-85
The Font Information Resource	8-86
The Control Device Function Code Resource	8-87
The Card Resource	8-87
The Monitor Code Resource	8-88
The Rectangle Resource	8-88
Summary of Control Panels	8-89
Pascal Summary	8-89
Constants	8-89
Application-Defined Routines	8-90
C Summary	8-90
Constants	8-90
Application-Defined Routines	8-92

Chapter 9 Desktop Manager 9-1

About the Desktop Database	9-4
Using the Desktop Manager	9-4
Desktop Manager Reference	9-6
Data Structure	9-6
The Desktop Parameter Block	9-7
Routines	9-8
Locating, Opening, and Closing the Desktop Database	9-9
Reading the Desktop Database	9-12
Adding to the Desktop Database	9-17
Deleting Entries From the Desktop Database	9-20
Manipulating the Desktop Database Itself	9-23
Summary of the Desktop Manager	9-27
Pascal Summary	9-27
Constants	9-27

Data Types	9-27	
Routines	9-28	
C Summary	9-30	
Constants	9-30	
Data Types	9-31	
Routines	9-31	
Assembly-Language Summary		9-34
Data Structures	9-34	
Trap Macros	9-35	
Result Codes	9-35	

Glossary GL-1

Index IN-1

Figures, Tables, and Listings

Figure 1-1	The data fork and resource fork of a file	1-4
Figure 1-2	An application's and a document's data fork and resource fork	1-6
Figure 1-3	Resource attributes	1-8
Figure 1-4	A typical search order for a specific resource	1-11
Figure 1-5	The ResEdit window for the SurfWriter application	1-15
Figure 1-6	The menus of the SurfWriter application	1-16
Figure 1-7	Getting a handle to a resource	1-19
Figure 1-8	A handle to a purgeable resource after the resource has been purged	1-20
Figure 1-9	Detaching a resource	1-23
Figure 1-10	Resource ID of an owned resource	1-48
Figure 1-11	Format of a resource fork	1-121
Figure 1-12	Format of a resource header in a resource fork	1-122
Figure 1-13	Format of resource data for a single resource	1-122
Figure 1-14	Format of the resource map in a resource fork	1-123
Figure 1-15	Format of an item in a resource type list	1-123
Figure 1-16	Format of an entry in the reference list for a resource type	1-124
Figure 1-17	Format of an item in a resource name list	1-124
Figure 1-18	Offsets in a resource fork and an entry for a single resource in a reference list	1-125
Figure 1-19	Structure of a compiled ROM override ('ROV#') resource	1-136
Table 1-1	Typical locations of resources	1-12
Table 1-2	Standard resource types	1-43
Table 1-3	Resource types reserved for use by system software	1-46
Table 1-4	Document and application icons	1-130
Table 1-5	Folder icons	1-131
Table 1-6	System Folder icons	1-132
Table 1-7	Desktop icons	1-133
Table 1-8	Standard File Package icons	1-133
Listing 1-1	A menu in Rez input format	1-17
Listing 1-2	Safely changing a resource that is purgeable	1-21
Listing 1-3	Releasing a resource	1-22
Listing 1-4	Detaching a resource	1-24
Listing 1-5	Getting the file reference number for your application's resource fork	1-25
Listing 1-6	Creating an empty resource fork	1-26
Listing 1-7	Creating and opening a resource fork	1-27
Listing 1-8	Saving and restoring the current resource file	1-29
Listing 1-9	Getting a resource from a document file	1-32
Listing 1-10	Counting and indexing through resources	1-34

Listing 1-11	Saving a resource to a resource fork	1-38
Listing 1-12	Using partial resource routines	1-41

Chapter 2

Scrap Manager 2-1

Figure 2-1	Copying and pasting data between two applications using the scrap	2-5
Figure 2-2	Writing both standard formats to the scrap	2-8
Figure 2-3	Using a private scrap	2-9
Figure 2-4	Intelligent cut and paste	2-11
Figure 2-5	Non-intelligent cut and paste	2-11
Figure 2-6	Location of the scrap in memory	2-13
Table 2-1	Actions your application performs in response to editing commands	2-6
Listing 2-1	Writing data to the scrap	2-16
Listing 2-2	Writing data to a private scrap	2-18
Listing 2-3	Copying data from the scrap in response to suspend events	2-19
Listing 2-4	Handling the Paste command using the scrap	2-21
Listing 2-5	Handling the Paste command using a private scrap	2-24
Listing 2-6	Handling resume events	2-25
Listing 2-7	Converting data between the scrap and a private scrap	2-27
Listing 2-8	Using TextEdit to handle the Cut command	2-29
Listing 2-9	Using TextEdit to handle the Paste command	2-30

Chapter 3

Help Manager 3-1

Figure 3-1	The Help menu for the Finder	3-7
Figure 3-2	A help balloon drawn with the standard balloon definition function	3-8
Figure 3-3	The tip and hot rectangle for a help balloon	3-9
Figure 3-4	Standard balloon positions and their variation codes	3-10
Figure 3-5	Alternate positions of a help balloon	3-11
Figure 3-6	Default help balloons for the window frame	3-15
Figure 3-7	Default help balloons for the Apple and Help menus	3-16
Figure 3-8	Default help balloons for application and document icons	3-17
Figure 3-9	Help balloons for different states of the Cut command	3-29
Figure 3-10	A help balloon for an enabled menu title	3-37
Figure 3-11	A help balloon for a dimmed menu title	3-37
Figure 3-12	A help balloon for a menu title dimmed by the Dialog Manager	3-38
Figure 3-13	A help balloon for menu items dimmed by the Dialog Manager	3-38
Figure 3-14	A help balloon for a menu item	3-39
Figure 3-15	A help balloon for a dimmed menu item	3-40
Figure 3-16	Help balloons for a changing menu item	3-45
Figure 3-17	A help balloon in a modal dialog box	3-61
Figure 3-18	Static and dynamic windows	3-64

Figure 3-19	A tool palette with a help balloon	3-70
Figure 3-20	A help balloon for a dialog box with a title	3-72
Figure 3-21	Default and custom help balloons for an application icon	3-86
Figure 3-22	The Help menu with an appended menu item	3-90
Figure 3-23	Structure of a compiled menu help ('hmnu') resource	3-133
Figure 3-24	Structure of an 'hmnu' component compiled with the HMStringItem identifier	3-135
Figure 3-25	Structure of an 'hmnu' component compiled with the HMStringResItem identifier	3-136
Figure 3-26	Structure of an 'hmnu' component compiled with the HMPictItem, HMTEResItem, or HMSTRResItem identifier	3-137
Figure 3-27	Structure of an 'hmnu' component compiled with the HMSkipItem identifier	3-138
Figure 3-28	Structure of a menu-item component compiled with the HMCompareItem identifier	3-139
Figure 3-29	Structure of a menu-item component compiled with the HMNamedResourceItem identifier	3-140
Figure 3-30	Structure of a compiled dialog-item help ('hdlg') resource	3-141
Figure 3-31	Structure of an 'hdlg' component compiled with the HMStringItem identifier	3-144
Figure 3-32	Structure of an 'hdlg' component compiled with the HMStringResItem identifier	3-145
Figure 3-33	Structure of an 'hdlg' component compiled with the HMPictItem, HMTEResItem, or HMSTRResItem identifier	3-146
Figure 3-34	Structure of an 'hdlg' component compiled with the HMSkipItem identifier	3-148
Figure 3-35	Structure of a compiled rectangle help ('hrct') resource	3-149
Figure 3-36	Structure of an 'hrct' component compiled with the HMStringItem identifier	3-150
Figure 3-37	Structure of an 'hrct' component compiled with the HMStringResItem identifier	3-151
Figure 3-38	Structure of an 'hrct' component compiled with the HMPictItem, HMTEResItem, or HMSTRResItem identifier	3-152
Figure 3-39	Structure of an 'hrct' component compiled with the HMSkipItem identifier	3-153
Figure 3-40	Structure of a compiled window help ('hwin') resource	3-155
Figure 3-41	Structure of a compiled Finder icon help ('hfdr') resource	3-157
Figure 3-42	Structure of an 'hfdr' component compiled with the HMStringItem identifier	3-158
Figure 3-43	Structure of an 'hfdr' component compiled with the HMStringResItem identifier	3-158
Figure 3-44	Structure of an 'hfdr' component compiled with the HMPictItem, HMTEResItem, or HMSTRResItem identifier	3-159
Figure 3-45	Structure of an 'hfdr' component compiled with the HMSkipItem identifier	3-160
Figure 3-46	Structure of a compiled default help override ('hovr') resource	3-161
Figure 3-47	Structure of an 'hovr' component compiled with the HMStringItem identifier	3-163
Figure 3-48	Structure of an 'hovr' component compiled with the HMStringResItem identifier	3-163
Figure 3-49	Structure of an 'hovr' component compiled with the HMPictItem, HMTEResItem, or HMSTRResItem identifier	3-164

Figure 3-50	Structure of an 'hovr' component compiled with the HMSkipItem identifier	3-165
Listing 3-1	Rez input for a partial 'hmenu' resource	3-31
Listing 3-2	Rez input for the missing-items component of an 'hmenu' resource	3-35
Listing 3-3	Rez input for corresponding 'hmenu' and 'STR#' resources	3-41
Listing 3-4	Rez input for an 'hmenu' resource that uses HMCompareItem for a changing menu item	3-44
Listing 3-5	Rez input for specifying help messages with named resources	3-46
Listing 3-6	Specifying an alternate 'hmenu' resource for a menu that your application disables when it displays movable modal dialog boxes	3-49
Listing 3-7	Reassigning 'hmenu' resources before displaying a movable modal dialog box	3-50
Listing 3-8	Rez input for an item list resource and an 'hdlg' resource	3-59
Listing 3-9	Rez input for corresponding 'hwin' and 'hrct' resources	3-71
Listing 3-10	Rez input for specifying help for titled and untitled windows	3-72
Listing 3-11	Using a string resource as the help message for HMShowBalloon	3-77
Listing 3-12	Using a picture resource as the help message for HMShowBalloon	3-77
Listing 3-13	Using a handle to a picture resource as the help message for HMShowBalloon	3-78
Listing 3-14	Using a string list resource as the help message for HMShowBalloon	3-79
Listing 3-15	Using styled text resources as the help message for HMShowBalloon	3-80
Listing 3-16	Using HMShowBalloon to display help balloons	3-82
Listing 3-17	Rez input for creating an 'hldr' resource for an application icon	3-86
Listing 3-18	Rez input for an 'hovr' resource	3-89
Listing 3-19	Rez input for specifying help balloons for items in the Help menu	3-91
Listing 3-20	Responding to the user's choice in a menu command	3-92
Listing 3-21	Using the HMExtractHelpMsg function	3-124
Listing 3-22	Using a tip function	3-131

Chapter 4

List Manager	4-1	
Figure 4-1	A one-column, text-only list without a scroll bar	4-4
Figure 4-2	A one-column, text-only list with a vertical scroll bar	4-5
Figure 4-3	A list whose scroll bar has been disabled	4-6
Figure 4-4	A deactivated list	4-6
Figure 4-5	A list containing multiple columns and graphical elements	4-7
Figure 4-6	A list of items whose cells display more than one type of information	4-8
Figure 4-7	A list with an item selected	4-9
Figure 4-8	Selection of a range of items in a list	4-10

Figure 4-9	Effect of dragging after Shift-clicking	4-11
Figure 4-10	Selection of discontinuous items in a list	4-12
Figure 4-11	Effect of Shift-clicking in a list that contains discontinuous items	4-13
Figure 4-12	Notifying the user of nonstandard list behavior	4-14
Figure 4-13	Response to pressing the Command–Up Arrow keys	4-16
Figure 4-14	Response to user making a discontinuous selection, then pressing Shift–Right Arrow followed by Shift–Left Arrow using the extend algorithm	4-17
Figure 4-15	Response to Shift–Right Arrow using the anchor algorithm	4-19
Figure 4-16	An outlined list in a window with more than one list	4-21
Figure 4-17	Coordinates of cells	4-22
Figure 4-18	Selection flags	4-38
Figure 4-19	The Chooser’s use of a custom list definition procedure	4-58
Listing 4-1	Creating a list with a vertical scroll bar	4-27
Listing 4-2	Installing a list in a dialog box	4-29
Listing 4-3	Drawing a border around a list	4-30
Listing 4-4	Adding items from a string list to a one-column, text-only list	4-31
Listing 4-5	Responding to a mouse-down event in a list	4-33
Listing 4-6	Responding to an update event in a list	4-33
Listing 4-7	Finding the first selected cell in a list	4-34
Listing 4-8	Finding the last selected cell in a list	4-35
Listing 4-9	Selecting a cell and deselecting other cells	4-36
Listing 4-10	Scrolling so that a particular cell is visible	4-37
Listing 4-11	Clearing all cell data	4-40
Listing 4-12	Getting a copy of the data of a cell	4-41
Listing 4-13	Directly accessing a cell’s data	4-41
Listing 4-14	Adding an item to a one-column, alphabetical text list	4-42
Listing 4-15	A match function	4-43
Listing 4-16	Searching a list for a cell containing certain text or the next cell alphabetically	4-44
Listing 4-17	Resetting variables related to type selection	4-46
Listing 4-18	Selecting an item in response to a key-down event	4-47
Listing 4-19	Determining the location of a new cell in response to an arrow-key event	4-49
Listing 4-20	Moving the selection in response to an arrow-key event	4-50
Listing 4-21	Extending the selection in response to an arrow-key event	4-51
Listing 4-22	Processing an arrow-key event	4-52
Listing 4-23	Drawing an outline around a list	4-54
Listing 4-24	Adding a list to the ring	4-55
Listing 4-25	Updating the outline of all lists in a window	4-56
Listing 4-26	Moving the outline to the next list in a window	4-57
Listing 4-27	Moving the outline to the previous list in a window	4-57
Listing 4-28	Processing messages to a list definition procedure	4-59
Listing 4-29	Using the default initialization method	4-60
Listing 4-30	Responding to the <code>lDrawMsg</code> message	4-61
Listing 4-31	Responding to the <code>lHiliteMsg</code> message	4-62
Listing 4-32	Responding to the <code>lCloseMsg</code> message	4-63
Listing 4-33	Setting the cell size of a list	4-63
Listing 4-34	Adding an icon to a list of icons	4-64

Chapter 5

Icon Utilities 5-1

Figure 5-1	The ResEdit view of an icon	5-4
Figure 5-2	An icon family	5-5
Listing 5-1	Drawing the icon from an icon family that is best suited to the user's display	5-10
Listing 5-2	Drawing the icon from an icon suite that is best suited to the display device	5-11
Listing 5-3	Drawing a specific icon from an icon family or icon suite	5-12
Listing 5-4	Manipulating icon data in memory	5-13
Listing 5-5	Drawing an icon of resource type 'ICON'	5-14
Listing 5-6	Drawing an icon of resource type 'ICON' with a specific alignment and transform	5-15
Listing 5-7	Drawing an icon of resource type 'icn'	5-15
Listing 5-8	Drawing an icon of resource type 'icn' with a specific alignment and transform	5-16
Listing 5-9	Drawing an icon of resource type 'SICN' with a specific alignment and transform	5-16

Chapter 6

Component Manager 6-1

Figure 6-1	The relationship between an application, the Component Manager, and components	6-5
Figure 6-2	Supporting multiple component connections	6-34
Figure 6-3	Interaction between the <code>componentFlags</code> and <code>componentFlagsMask</code> fields	6-40
Figure 6-4	Format of a component file	6-84
Figure 6-5	Structure of a compiled component ('thng') resource	6-85
Table 6-1	Request codes	6-14
Listing 6-1	Finding a component	6-9
Listing 6-2	Opening a specific component	6-10
Listing 6-3	Getting information about a component	6-10
Listing 6-4	Using a drawing component	6-11
Listing 6-5	A drawing component for ovals	6-16
Listing 6-6	Responding to an open request	6-20
Listing 6-7	Responding to a close request	6-21
Listing 6-8	Responding to the can do request	6-22
Listing 6-9	Responding to the setup request	6-26
Listing 6-10	Responding to the draw request	6-27
Listing 6-11	Responding to the erase request	6-27
Listing 6-12	Responding to the click request	6-27
Listing 6-13	Responding to the move to request	6-28
Listing 6-14	Registering a component	6-31
Listing 6-15	Rez input for a component resource	6-33
Listing 6-16	Delegating a request to another component	6-36

Chapter 7

Translation Manager 7-1

Figure 7-1	The Finder's application-unavailable alert box	7-5
Figure 7-2	The application-unavailable alert box for 'TEXT' and 'PICT' documents	7-5
Figure 7-3	The translation choices dialog box	7-6
Figure 7-4	A translation progress dialog box	7-7
Figure 7-5	The modified application-unavailable alert box	7-7
Figure 7-6	The enhanced file-opening dialog box	7-8
Figure 7-7	Document Converter configuration dialog box	7-9
Figure 7-8	A translation group with multiple source and destination types	7-29
Figure 7-9	A translation group with a single destination type	7-29
Figure 7-10	Point-to-point translation	7-30
Figure 7-11	Structure of a compiled open ('open') resource	7-44
Figure 7-12	Structure of a compiled kind ('kind') resource	7-45
Listing 7-1	Translation-specific selectors and response bit for Gestalt	7-12
Listing 7-2	A sample resource of type 'open'	7-13
Listing 7-3	A sample resource of type 'kind'	7-15
Listing 7-4	Sample resources for a translation extension	7-22
Listing 7-5	Handling Component Manager request codes	7-25
Listing 7-6	Creating a file translation list	7-30
Listing 7-7	Identifying file types	7-33
Listing 7-8	Translating a document	7-34

Chapter 8

Control Panels 8-1

Figure 8-1	Two control panels, each with its own window	8-5
Figure 8-2	The General Controls control panel	8-6
Figure 8-3	Control panel icons in the Control Panels folder	8-9
Figure 8-4	The Monitors control panel	8-10
Figure 8-5	An Options dialog box for the SurfBoard video card	8-11
Figure 8-6	The River control panel interface	8-13
Figure 8-7	An icon for the River control panel file	8-14
Figure 8-8	The Color control panel	8-15
Figure 8-9	Coordinates defining the rectangles of the River control panel display area	8-16
Figure 8-10	Example of an inactive control panel	8-34
Figure 8-11	An Options dialog box with standard controls	8-49
Figure 8-12	An Options dialog box with superuser controls	8-50
Figure 8-13	The SurfBoard monitors extension icon	8-51
Figure 8-14	Display area defined by a rectangle resource	8-53
Figure 8-15	The SurfBoard Options dialog box with superuser controls	8-54
Figure 8-16	Structure of a compiled machine ('mach') resource	8-84
Figure 8-17	Structure of a compiled rectangle positions ('nrect') resource	8-85
Figure 8-18	Structure of a compiled font information ('finf') resource	8-86
Figure 8-19	Structure of a compiled card ('card') resource	8-87
Figure 8-20	Structure of a compiled rectangle ('RECT') resource	8-88

Table 8-1	Possible settings for the machine resource masks	8-21
Table 8-2	Error codes and their meaning	8-47
Table 8-3	Messages from the Finder	8-76
Table 8-4	Messages from the Monitors control panel	8-80
Table 8-5	Possible settings for the machine resource masks	8-85
Listing 8-1	Rez input for a rectangle positions list ('nrct') resource	8-16
Listing 8-2	Rez input for an item list ('DITL') resource	8-18
Listing 8-3	Rez input for a machine ('mach') resource	8-21
Listing 8-4	Rez input for a file reference ('FREF') resource	8-21
Listing 8-5	Rez input for a signature resource	8-22
Listing 8-6	Rez input for a bundle ('BNDL') resource	8-22
Listing 8-7	A control panel's static text defined as user items	8-24
Listing 8-8	A control device function	8-27
Listing 8-9	Initializing a control panel: Allocating memory and setting controls	8-31
Listing 8-10	Responding to an activate event	8-35
Listing 8-11	Responding to a keyboard event	8-38
Listing 8-12	Responding to the user's interaction with controls	8-41
Listing 8-13	Responding to update events	8-43
Listing 8-14	Drawing text defined as user items	8-44
Listing 8-15	Terminating a control device function when the user closes the control panel	8-45
Listing 8-16	Responding to Edit menu commands	8-46
Listing 8-17	Rez input for a card ('card') resource	8-52
Listing 8-18	Rez input for a rectangle ('RECT') resource	8-53
Listing 8-19	Rez input for the SurfBoard monitors extension item list resource	8-55
Listing 8-20	Rez input for icon family resources for a monitors extension	8-57
Listing 8-21	Rez input for a version ('vers') resource	8-58
Listing 8-22	Rez input for the SurfBoard string list resource	8-59
Listing 8-23	Rez input for a file reference resource of a monitors extension	8-60
Listing 8-24	Rez input for a bundle resource of a monitors extension	8-60
Listing 8-25	A monitors extension function	8-64
Listing 8-26	Handling the startup message	8-66
Listing 8-27	Using a normal user rectangle or extending it to display superuser controls	8-67
Listing 8-28	Initializing a monitors extension	8-69
Listing 8-29	Drawing a line to separate superuser controls	8-70
Listing 8-30	Responding when a user clicks a control	8-72

About This Book

This book, *Inside Macintosh: More Macintosh Toolbox*, together with the book *Inside Macintosh: Macintosh Toolbox Essentials*, describes features you can build into your Macintosh application and documents the system software routines for implementing those features.

For information about events, windows, menus, controls, alert boxes, and dialog boxes and about how your application interacts with the Finder, see *Inside Macintosh: Macintosh Toolbox Essentials*.

This book, *More Macintosh Toolbox*, describes how you can enhance your application by supporting copy and paste and providing messages for help balloons. In addition, it describes other features you may want to use in your application, such as scrolling lists in dialog boxes and icons in windows. It also explains how to create resources, components, translation extensions, and control panels.

To read and write resources, see the chapter “Resource Manager.” This chapter describes how you can use resources to store the descriptions of user interface elements such as menus, windows, controls, dialog boxes, and icons. You can also use resources to store variable settings, such as the location of the window at the time the user closes it. When the user opens the document again, your application can read the information in the resource and restore the window to its previous location.

To support copy-and-paste operations in your application, see the chapter “Scrap Manager.” By using the Scrap Manager, you can allow users to copy and paste data between documents created by your application and documents created by other applications.

To provide messages for help balloons for elements of your application, see the chapter “Help Manager.” Help balloons are rounded-rectangle windows that contain explanatory information for the user. With tips pointing at the objects they annotate, help balloons look like the bubbles used for dialog in comic strips. Help balloons are turned on by the user from the Help menu; when Balloon Help assistance is on, a help balloon appears whenever the user moves the cursor over the balloon’s hot rectangle.

To create lists in your application’s dialog boxes, including lists that contain scroll bars, see the chapter “List Manager.” You can use the List Manager to create one-column or multicolumn lists. Lists are useful for allowing the user to select one or more items from a group of items.

To display icons in a window or dialog box of your application, see the chapter “Icon Utilities.” By using Icon Utilities routines, you can automatically draw the icon from an icon family that is best suited for the current bit depth of the monitor.

PREFACE

To use or create components, see the chapter “Component Manager.” Components can provide your application with various services such as image compression or decompression services. You can also provide services to other applications by creating your own component.

To direct the translation of documents from one format to another, see the chapter “Translation Manager.” Macintosh Easy Open uses the Translation Manager to automatically provide some translation services for your application. Optionally, you can enhance your application’s interaction with Macintosh Easy Open or provide your own translation services.

To create a control panel or an extension to the Monitors control panel, see the chapter “Control Panels.” Control panels allow the user to set preferences for systemwide features, such as the speaker volume, desktop pattern, or picture displayed by a screen saver. Extensions to the Monitors control panel should be created only by the manufacturer of a video device.

To get information from the desktop database, see the chapter “Desktop Manager.” The desktop database contains information used by the Finder, such as icon definitions and their associated file types, as well as any comments that the user has added to the information window for desktop objects.

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

Some related topics can be found in other *Inside Macintosh* books. For information on how to read and write to the data fork of a file, see the chapter “Introduction to File Management” in *Inside Macintosh: Files*. For information about drawing into a window or other graphics port, see *Inside Macintosh: Imaging with QuickDraw*. 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 chapter “Resource Manager” contains these sections:

- “Introduction to Resources.” This section presents a general introduction to resources, resource types, and resource forks.
- “About the Resource Manager.” This section provides an overview of the features provided by the Resource Manager.
- “Using the Resource Manager.” This section describes the tasks you can accomplish using the Resource Manager. It describes how to use the most common routines, gives related user interface information, provides code samples, and supplies additional information.

PREFACE

- “Resource Manager Reference.” This section provides a complete reference to the Resource 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 Resource Manager.” This section provides the Pascal and C interfaces for the constants, data structures, routines, and result codes associated with the Resource 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.

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 not essential to an understanding of the main text. (An example appears on page 1-9.) ◆

IMPORTANT

A note like this contains information that is essential for an understanding of the main text. (An example appears on page 1-5.) ▲

▲ 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 1-5.) ▲

PREFACE

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

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

PREFACE

code samples in your application. You can find the location of code listings in the list of figures, tables, and listings beginning on page xvii. If you know the name of a particular routine (such as `DoPicTBalloon` or `MyPlotAnICON`) 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.

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 referring to a sample application called *SurfWriter*; this book also refers to the sample applications *SurfPaint* and *SurfDB*. These applications are not actual products of Apple Computer, Inc. This book also refers to a River control panel and SurfBoard display card. These are not actual products of Apple Computer, Inc. In addition, the name River Change Systems is used to represent a fictitious company.

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.

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

AppleLink: APDA

America Online: APDA

CompuServe: 76666,2405

Internet: APDA@applelink.apple.com

P R E F A C E

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, and other technical information, contact

Macintosh Developer Technical Support
Apple Computer, Inc.
20525 Mariani Avenue, M/S 303-2T
Cupertino, CA 95014-6299