



Macintosh Human Interface Guidelines

by Apple Computer, Inc.





Macintosh Human Interface Guidelines



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, AppleLink, AppleShare, AppleTalk, EtherTalk, HyperTalk, ImageWriter, LaserWriter, Macintosh, MultiFinder, and StyleWriter are trademarks of Apple Computer, Inc., registered in the United States and other countries.

BalloonHelp, BalloonWriter, Finder, PowerBook, QuickDraw, 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 and Palatino are registered trademarks of Linotype Company.

HyperCard, MacDraw, MacPaint, and MacWrite are registered trademarks of Claris Corporation.

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

Varityper is a registered trademark of Varityper, Inc.

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 0-201-62216-5
1 2 3 4 5 6 7 8 9-BA-9695949392
First Printing, November 1992



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

Contents

Figures and Tables xiii

Preface **About This Book** xxi

Who Should Read This Book xxi
What's New in Macintosh Human Interface From Apple xxii
About *Making It Macintosh* xxii
What's in This Book xxii
 The Basic Philosophy xxii
 The Interface Elements xxiii
 Appendixes xxiii
Visual Cues Used in This Book xxiii

Part 1 **Fundamentals** 1

Chapter 1 **Human Interface Principles** 3

The Human Interface Design Principles 4
 Metaphors 4
 Direct Manipulation 5
 See-and-Point 7
 Consistency 7
 WYSIWYG (What You See Is What You Get) 8
 User Control 9
 Feedback and Dialog 9
 Forgiveness 10
 Perceived Stability 11
 Aesthetic Integrity 11
 Modelessness 12
Additional Issues to Consider 13
 Knowledge of Your Audience 13
 Accessibility 14

Chapter 2 **General Design Considerations** 15

Worldwide Compatibility 16
 Cultural Values 17
 Resources 17

Language Differences	18
Text Display and Text Editing	19
Default Alignment of Interface Elements	20
Keyboards	22
Fonts	23
Universal Access	24
People With a Physical Disability	25
People With a Visual Disability	25
People With a Hearing Disability	26
People With a Speech or Language Disability	27
People With a Seizure Disorder	27
Collaborative Computing	27
Concern for Other Users	28
User Identification	28
Access Privileges	28
Passwords	29
Data Encryption for Security	30
Clear Communications	30
Displaying the Current State of Data	30
Communicating With Other Environments	31
Network Transparency	31

Chapter 3 Human Interface Design and the Development Process 33

Design Decisions	34
Features Inspired by Market Pressures	34
Feature Cascade	35
The 80 Percent Solution	35
Managing Complexity	35
Using Progressive Disclosure	35
Implementing Preferences	37
Extending the Interface	38
When to Go Beyond the Guidelines	38
Build on the Existing Interface	39
Don't Assign New Behaviors to Existing Objects	39
Create a New Interface Element Cautiously	40
Involving Users in the Design Process	41
Define Your Audience	41
Analyze Tasks	41
Build Prototypes	42
Observe Users	42
Ten Steps for Conducting a User Observation	43

The Menu Bar	52
Menu Behavior	55
Menu Elements	58
Menu Item Names	58
Grouping Items in Menus	60
Menu Dividers	62
Standard Characters and Text Style in Menus	64
Checkmarks and Dashes in Menus	64
The Ellipsis Character in Menus	67
A Diamond Mark in the Application Menu	71
Avoid Nonstandard Marks in Menus	72
Text Styles in Menus	73
Toggled Menu Items	75
Scrolling Menus	78
Hierarchical Menus	79
Pop-Up Menus	82
Standard Pop-Up Menus	87
Type-In Pop-Up Menus	91
Tear-Off Menus and Palettes	92
Tear-Off Menus	93
Palettes	96
Standard Macintosh Menus	98
The Apple Menu	98
About	98
File Menu	99
New	99
Open	101
Close	102
Save	104
Save As	106
Revert	107
Page Setup...	108
Print...	108
Quit	109
The Edit Menu	109
The Clipboard	111
Undo/Redo	113
Cut	114
Copy	115
Paste	115
Clear	117

Select All	117
Show Clipboard/Hide Clipboard	117
Create Publisher...	117
Subscribe To...	118
Publisher/Subscriber Options...	118
The Font Menu	120
The Size Menu	122
The Style Menu	124
The Help Menu	125
The Keyboard Menu	125
The Application Menu	127
Keyboard Equivalents	128

Chapter 5

Windows 131

Window Appearance	134
Document Window Controls	134
Use of Color in Windows	135
Utility Windows	137
Window Behaviors	139
The Active Window	139
Opening Windows	141
Window Display Order	143
Window Positions	146
The Default Position on a Single Screen	147
The Default Position on Multiple Screens	148
Dialog Box and Alert Box Positions	150
Closing a Window	152
Moving a Window	154
Changing the Size of a Window	156
Scrolling a Window	158
Scroll Bars	158
Scrolling With the Scroll Arrows	163
Scrolling With the Gray Area	164
Scrolling by Dragging the Scroll Box	164
Automatic Scrolling	166
The Zoom Box and Window Behavior	168
Splitting a Window	170
Window Pane Behavior	172
One Split per Orientation	173

Chapter 6

Dialog Boxes 175

Modeless Dialog Boxes	178
Modeless Dialog Box Appearance	179
Modeless Dialog Box Behaviors	181
Menu Bar Access	181
Accepting Changes in a Modeless Dialog Box	182
Completing Commands	184
Movable Modal Dialog Boxes	185
Movable Modal Dialog Box Appearance	186
Movable Modal Dialog Box Behaviors	187
Menu Bar Access	187
Modal Dialog Boxes	188
Modal Dialog Box Appearance	190
Modal Dialog Box Behaviors	191
Menu Bar Access	191
Stacking Modal Dialog Boxes	192
Alert Boxes	193
Alert Box Appearance	194
Note Alert Boxes	194
Caution Alert Boxes	195
Stop Alert Boxes	196
Basic Dialog Box Layout	196
Keyboard Navigation in Dialog Boxes	198
Dialog Box Messages	199
Standard File Dialog Boxes	200
Save Changes Alert Box	201

Chapter 7

Controls 203

Standard Toolbox Controls	204
Buttons	204
Button Behavior	205
Button Names	206
Radio Buttons	210
Checkboxes	211
Controls Not Supported by the Macintosh Toolbox	214
Sliders	214
Little Arrows	216
Outline Triangles	218
Other Elements for User Interaction	218
Text Entry Fields	219
Scrolling Lists	220

Chapter 8

Icons 223

Why Icons Work	224
Limitations of Icons	227
Designing Effective Icons	229
Use Appropriate Metaphors	229
Think About Worldwide Compatibility	230
Avoid Text in Icons	230
Design for the Macintosh Display	231
Use a Consistent Light Source	232
Optimize for Your Target Display	232
Maintain a Consistent Visual Appearance in an Icon Family	233
Use Icon Elements Consistently	233
The Finder Icon Family	234
An Icon Design Process	236
Black-and-White Icons	238
Color Icons	238
Icon Colors	240
The Apple Icon Color Set	240
Degradation of the Color Set Across Monitors	241
Selection Mechanism for Color Icons	241
Color Labeling Mechanism for Color Icons	242
Anti-Aliasing	243
Small Icons	244
Default and Custom Icons	245
Application Icons	246
Document Icons	247
Stationery Pad Icons	248
Query Document Icons	249
Edition Icons	250
Preferences Icons	250
Extension Icons	250
Control Panel Icons	251
Movable Resource Icons	252
Keyboard Icons	252

Chapter 9

Color 257

Color Design of Standard Interface Elements	258
Windows and Dialog Boxes	258
Menus	260
Pointers	260
Highlighting and Selection	260

Color Application Guidelines	261
Match Complexity to the Level of User	261
Design for the Macintosh	262
Design for Black and White First	263
Limit the Number of Colors	264
Colors on Gray	265
Beware of Blue	265
Small Objects	265
Color for Categorizing Information	265

Chapter 10

Behaviors 267

The Pointing Device	268
Mouse Actions	271
Clicking	271
Double-Clicking	272
Pressing	273
Dragging	274
The Keyboard	275
Character Keys	275
Enter	275
Tab	276
Return	276
Delete (or Backspace)	277
Clear	277
Escape	277
Modifier Keys	278
Shift	278
Caps Lock	279
Option	279
Command	280
Control	280
Type-Ahead and Auto-Repeat	280
International Keyboards	281
Arrow Keys	281
Appropriate Uses for the Arrow Keys	281
Moving the Insertion Point	282
Moving the Insertion Point in Empty Documents	282
Using Modifier Keys With Arrow Keys	282
Function Keys	284
Help	285
Forward Delete (Del)	285
Home	285
End	285
Page Up	286
Page Down	286

Selecting	286
Selection Methods	288
Selection by Clicking	289
Selection by Dragging	289
Changing a Selection With Shift-Click	289
Changing a Selection With Command-Click	291
Selections in Text	292
Selecting With the Mouse	293
Selecting Ranges	294
Selecting With the Arrow Keys	295
Selections in Graphics	297
Selections in Arrays and Tables	298
Editing Text	300
Inserting Text	300
Deleting Text	300
Replacing a Selection	301
Intelligent Cut and Paste	301
Editing Fields	302

Chapter 11

Language 305

Style	306
Terminology	307
Developer Terms and User Terms	307
Terms That Are Often Misused	308
Click	308
Checkbox	308
Document	308
File	308
Utility Window	309
Labels for Interface Elements	309
Dialog Box Messages	310
User Documentation	313
Online Help Systems	314
Provide Concurrent Help	314
Provide Multiple Levels of Help	314
Assist Users by Answering Their Questions	315
Keep the Help System Simple	316
Design Online Help as an Interactive Coach	316
Balloon Help	316
When to Use a Help Balloon	317
How to Write a Balloon	318
Wording for Specific Balloon Types	319
Buttons With Words	319
Menu Titles	320
Menu Items	320

Radio Buttons	321
Checkboxes	321
Groups of Checkboxes or Radio Buttons	322
Tools in Palettes	323
Window Parts	324
Modal Dialog Box on the Screen	324
Icons	324
Text Entry Boxes	325

Appendixes 327

Appendix A Resources 329

Association for Computing Machinery (ACM)	329
<i>Communications of the ACM</i>	329
SIGCHI	329
SIGGRAPH	330
CSCW	330
Human Factors Society	331
Human Factors Society Annual Meeting	331
<i>Human Factors</i>	331
<i>Human Factors Society Bulletin</i>	332
Apple Developer Information	332
APDA	332
Developer Support Center	333
In-House Development Support	333
<i>develop</i>	333

Appendix B Bibliography 335

Animation	336
Cognitive Psychology and Human Factors	336
Color	337
Environmental Design	338
Graphic and Information Design	339
Graphic Design and Drawing	339
Icons and Symbols	339
Typography	340
History of Human Interface	340
Human-Computer Design	341
Consistency	341
Direct Manipulation	341

Menus	342
Metaphors	342
Product Design	343
Usability Testing	343
User-Centered Design	344
Human-Computer Interaction	344
Language	346
Programming	346
Special Applications	347
Collaborative Computing	347
Hypertext	347
Multimedia	348
Online Documentation and Online Help	348
Universal Access	349
Visual Thinking	349
Worldwide Software	350

Appendix C	Checklist	351
	General Considerations	351
	Graphic Design	353
	Color	353
	Icons	354
	Windows	354
	Dialog Boxes	355
	Alert Boxes	357
	Scrolling	357
	Menus	358
	Pop-Up Menus	359
	Palettes and Tear-Off Menus	359
	Mouse Standards	360
	Text	360
	Balloon Help	360
	Keyboard Equivalents	361
	Edition Manager	361
	Documentation	362

Glossary	363
----------	-----

Index	373
-------	-----

Figures and Tables

Chapter 1	Human Interface Principles	3
	Figure 1-1	Direct manipulation 6
	Figure 1-2	An example of a bad message and an example of a helpful message 10
	Figure 1-3	Don't use arbitrary graphic elements 12
Chapter 2	General Design Considerations	15
	Figure 2-1	Menu bars in different languages 18
	Figure 2-2	English and Arabic dialog boxes 20
	Figure 2-3	Dialog boxes with display rectangles that are different sizes and the same size 21
	Figure 2-4	Right-to-left alignment of dialog box items 21
	Figure 2-5	The Keyboard menu 23
	Figure 2-6	The boundaries of a font 24
	Figure 2-7	The Sound control panel 26
	Figure 2-8	A shutdown message 28
	Figure 2-9	The AppleShare connect dialog box 29
Chapter 3	Human Interface Design and the Development Process	33
	Figure 3-1	An expanding dialog box 36
	Figure 3-2	Directions a window can expand 37
	Figure 3-3	An incorrect subpalette indicator 40
	Figure 3-4	A better subpalette indicator 40
Chapter 4	Menus	49
	Figure 4-1	The standard order of actions 51
	Figure 4-2	A menu bar 53
	Figure 4-3	Three menu bars 53
	Figure 4-4	The Finder menu bar in six languages 54
	Figure 4-5	An unavailable menu 55
	Figure 4-6	Opening a menu 56
	Figure 4-7	A feedback technique 57
	Figure 4-8	A typical menu 58
	Figure 4-9	A menu with adjectives 59
	Figure 4-10	Command names properly capitalized 59
	Figure 4-11	Unavailable items aren't highlighted 60
	Figure 4-12	Menus with appropriate groups 61
	Figure 4-13	Grouping items in menus 62
	Figure 4-14	Standard menu dividers 63
	Figure 4-15	An inappropriate menu divider 63

Figure 4-16	A menu with text styles and an indicator	64
Figure 4-17	A checkmark to indicate a choice in a mutually exclusive group	65
Figure 4-18	A checkmark to indicate a choice in an accumulating attribute group	65
Figure 4-19	Dashes to indicate partial attributes in an accumulating attribute group	66
Figure 4-20	Several attributes in effect	67
Figure 4-21	The ellipsis character means more information is required	68
Figure 4-22	Don't use the ellipsis character with a command that doesn't require more information	69
Figure 4-23	The absence of the ellipsis character means no more information is required	70
Figure 4-24	The ellipsis character doesn't mean an alert box appears	71
Figure 4-25	The Application menu with a notification symbol	72
Figure 4-26	Don't use arbitrary symbols in menus	72
Figure 4-27	A Style menu with text styles	73
Figure 4-28	The effects of the two states of a Style menu item	74
Figure 4-29	A menu with nonstandard marks and extraneous text styles and a menu all in plain text style	75
Figure 4-30	A set of toggled menu items	76
Figure 4-31	A single toggled menu item whose name changes	76
Figure 4-32	An ambiguous toggled menu item	77
Figure 4-33	An incorrect use of a checkmark to indicate a state	78
Figure 4-34	A scrolling menu	78
Figure 4-35	The menu scrolling in the other direction	79
Figure 4-36	A hierarchical menu	79
Figure 4-37	Don't use submenus unnecessarily	80
Figure 4-38	A menu bar on a 9-inch screen with space for more menu titles	81
Figure 4-39	Examples of submenu titles	81
Figure 4-40	Avoid more than one level of submenus	82
Figure 4-41	A pop-up menu and its parts	83
Figure 4-42	Opening a pop-up menu	84
Figure 4-43	Pop-up menus versus radio buttons	85
Figure 4-44	Pop-up menus versus checkboxes	86
Figure 4-45	Don't use pop-up menus for commands	87
Figure 4-46	A standard pop-up menu	87
Figure 4-47	Using a pop-up menu	88
Figure 4-48	Correct and incorrect use of fonts in pop-up menus	89
Figure 4-49	Pop-up menu behavior	90
Figure 4-50	A hidden pop-up menu	91
Figure 4-51	A type-in pop-up menu	92
Figure 4-52	A type-in pop-up menu with user's choice added	92
Figure 4-53	A tools palette and a color palette	93
Figure 4-54	Using a tear-off menu	94
Figure 4-55	A tear-off menu on top of a document window	95
Figure 4-56	Palettes and feedback	96
Figure 4-57	A tool palette with the corresponding pointers	97
Figure 4-58	A tool palette in a window	97
Figure 4-59	An Apple menu	98
Figure 4-60	An About dialog box for an application	99

Figure 4-61	A File menu	99
Figure 4-62	The New command	100
Figure 4-63	The standard file dialog box for opening files	101
Figure 4-64	The save changes alert box	103
Figure 4-65	The correct location of the save changes alert box	104
Figure 4-66	The Save command	105
Figure 4-67	A sample alert box to use when a disk is full	105
Figure 4-68	The Save As command and dialog box	106
Figure 4-69	The Revert command	107
Figure 4-70	A Page Setup dialog box	108
Figure 4-71	A Print dialog box	109
Figure 4-72	A standard Edit menu for an application	110
Figure 4-73	Adding commands to the Edit menu	110
Figure 4-74	A sample Edit menu with Edition Manager commands	111
Figure 4-75	A sample hierarchical Edit menu with Edition Manager commands	111
Figure 4-76	The Clipboard	112
Figure 4-77	The Undo and Redo commands	114
Figure 4-78	The results of using the Paste command	116
Figure 4-79	The Create Publisher command and dialog box	118
Figure 4-80	The Subscribe To command and dialog box	118
Figure 4-81	The Publisher Options dialog box	119
Figure 4-82	The Subscriber Options dialog box	119
Figure 4-83	A Font menu	121
Figure 4-84	Don't combine the Font menu with other menus	122
Figure 4-85	A Size menu	122
Figure 4-86	A sample pull-down Size menu and font size dialog box	123
Figure 4-87	A Style menu	124
Figure 4-88	The Help menu	125
Figure 4-89	The Keyboard menu	126
Table 4-1	Apple-reserved keyboard equivalents for all systems	128
Table 4-2	Additional reserved keyboard equivalents for worldwide systems	128
Table 4-3	Common keyboard equivalents that are not reserved	129

Chapter 5

Windows 131

Figure 5-1	Examples of standard windows	133
Figure 5-2	Standard document window parts	134
Figure 5-3	Windows on a color screen	135
Figure 5-4	Standard window components in color	136
Figure 5-5	Colors that the user can choose for windows	137
Figure 5-6	A utility window	137
Figure 5-7	Make it clear where text will appear	138
Figure 5-8	The active window	139
Figure 5-9	Don't show a selection in an inactive window	141
Figure 5-10	Appropriate window titles for a series of unnamed windows	142
Figure 5-11	Examples of correct and incorrect window titles	143

Figure 5-12	Display order of document windows and modeless dialog boxes	144
Figure 5-13	Adding floating windows to the desktop	145
Figure 5-14	Adding a movable modal dialog box to the desktop	146
Figure 5-15	Window positions on a single screen	147
Figure 5-16	The standard window position on two sizes of screens	148
Figure 5-17	The standard window position on multiple screens	149
Figure 5-18	A window displayed across two screens	150
Figure 5-19	Standard position of an alert box	151
Figure 5-20	Alert box position in relation to the active document window	151
Figure 5-21	Standard alert box position with more than one screen	152
Figure 5-22	The close box	153
Figure 5-23	The save changes alert box	154
Figure 5-24	Moving a window	155
Figure 5-25	Multiple monitors and conceptual work space	156
Figure 5-26	A window growing larger	157
Figure 5-27	Relationship between a window and a document	158
Figure 5-28	The elements of a scroll bar	159
Figure 5-29	Using scroll arrows and the scroll box	159
Figure 5-30	Inactive scroll bars in active and inactive document windows	160
Figure 5-31	Background between the content and the window frame	161
Figure 5-32	Acceptable additions to the scroll bar region	162
Figure 5-33	Too many controls in the scroll bar	162
Figure 5-34	Scrolling by clicking a scroll arrow	163
Figure 5-35	Scrolling by clicking in the gray area	164
Figure 5-36	Scrolling by dragging the scroll box	165
Figure 5-37	Automatic scrolling	166
Figure 5-38	The zoom box	168
Figure 5-39	The standard state and the user state of a document	169
Figure 5-40	A split window	171
Figure 5-41	Split bar size	171
Figure 5-42	Independent and locked scrolling of window panes	172

Chapter 6

Dialog Boxes 175

Figure 6-1	Examples of dialog box types	177
Figure 6-2	A typical modeless dialog box	178
Figure 6-3	Two open modeless dialog boxes	179
Figure 6-4	The essential elements of a modeless dialog box	180
Figure 6-5	Incorrect absence of a close box in a modeless dialog box	181
Figure 6-6	Provide a place for the user to enter information in a modeless dialog box	184
Figure 6-7	A typical movable modal dialog box	185
Figure 6-8	The essential elements of a movable modal dialog box	186
Figure 6-9	Close box used incorrectly in a movable modal dialog box	187
Figure 6-10	A Finder movable modal dialog box	187
Figure 6-11	Menu bar access while a movable modal dialog box is open	188
Figure 6-12	An example of a modal dialog box	189
Figure 6-13	A status dialog box	190

Figure 6-14	The essential elements of a modal dialog box	190
Figure 6-15	Access to the Edit menu when displaying a modal dialog box	191
Figure 6-16	Second modal dialog box on top of first one	193
Figure 6-17	The essential elements of an alert box	194
Figure 6-18	An example of a note alert box	195
Figure 6-19	An example of a caution alert box	195
Figure 6-20	An example of a stop alert box	196
Figure 6-21	Recommended spacing of buttons and text in dialog and alert boxes	197
Figure 6-22	An active scrolling list	198
Figure 6-23	A well-written dialog box message	199
Figure 6-24	The standard file dialog box for opening files	200
Figure 6-25	The save changes alert box	201

Chapter 7

Controls 203

Figure 7-1	Buttons in a dialog box	205
Figure 7-2	A highlighted button	205
Figure 7-3	A dialog box with OK and Cancel buttons	207
Figure 7-4	A dialog box with a Done button instead of an OK button	208
Figure 7-5	A progress indicator that uses a Stop button	209
Figure 7-6	A confirmation alert box with appropriately named button	209
Figure 7-7	Sets of radio buttons	210
Figure 7-8	Radio buttons for selecting the alignment of text	211
Figure 7-9	The General Controls panel	211
Figure 7-10	A set of checkboxes	212
Figure 7-11	A single checkbox in a dialog box	212
Figure 7-12	The Find dialog box	213
Figure 7-13	An example of a slider	214
Figure 7-14	A slider with direction information	215
Figure 7-15	Incorrect use of a scroll bar and correct use of a slider	215
Figure 7-16	Little arrows control	216
Figure 7-17	Content-dependent increment	217
Figure 7-18	Outline triangle control	218
Figure 7-19	A text entry field	219
Figure 7-20	A scrolling list	220

Chapter 8

Icons 223

Figure 8-1	Common icons	224
Figure 8-2	Examples of common traffic symbols	225
Figure 8-3	Examples of commonly-used international symbols	225
Figure 8-4	Symbols are easier to understand than keyboard commands	226
Figure 8-5	Grouping icons on the desktop	226
Figure 8-6	A confusing image	227
Figure 8-7	Context clarifies the image	227
Figure 8-8	Icons with label text	228
Figure 8-9	A logical and an illogical metaphor	229

Figure 8-10	Localized mailbox icons	230
Figure 8-11	Avoid text in icons	231
Figure 8-12	Certain shapes don't work well	231
Figure 8-13	A consistent light source	232
Figure 8-14	Inconsistent light sources	232
Figure 8-15	Design the large icon first and base the small icon design on it	233
Figure 8-16	Consistent use of icon elements	234
Figure 8-17	An icon family	234
Figure 8-18	Different sizes of icons	235
Figure 8-19	A well-designed icon and its selected version	238
Figure 8-20	A poorly designed icon and its selected version	238
Figure 8-21	Icons with a black outline	239
Figure 8-22	Icons without a black outline	239
Figure 8-23	Standard 256-color palette with icon colors marked	240
Figure 8-24	An example of dithered color in an icon	241
Figure 8-25	Color icons and their selected states	242
Figure 8-26	Color icons and their color-labeled states	243
Figure 8-27	Correct anti-aliasing	243
Figure 8-28	Consistently designed small icons	244
Figure 8-29	Inconsistently designed small icons	245
Figure 8-30	Default application icons	246
Figure 8-31	Custom application icons	246
Figure 8-32	Examples of bad application icons	247
Figure 8-33	Default document icons	247
Figure 8-34	Application icon and document icon with the same graphic element	247
Figure 8-35	Acceptable and unacceptable custom document icons	248
Figure 8-36	Document icons with standard symbols	248
Figure 8-37	Default stationery pad icons	249
Figure 8-38	Default query document icons	249
Figure 8-39	Default edition icons	250
Figure 8-40	Preferences file icons	250
Figure 8-41	Default extension icons	251
Figure 8-42	Examples of Chooser icons	251
Figure 8-43	Icons for the Color control panel	251
Figure 8-44	Font icons	252
Figure 8-45	A sound icon	252
Figure 8-46	The default keyboard layout and input method icons	253
Figure 8-47	Examples of keyboard icons	253
Figure 8-48	Examples of modification indicators on keyboard icons	254
Figure 8-49	Enlarged keyboard icons with correct color substitutions	255
Table 8-1	Icon display on monitors of different bit depths	235
Table 8-2	Pattern substitutions for colors in keyboard icons	254

Chapter 9

Color 257

- Figure 9-1** A colorized window 259
Figure 9-2 A colorized movable modal dialog box 259
Figure 9-3 Color palette and custom color mixing tool 262
Figure 9-4 Design for black-and-white monitors first 263
Figure 9-5 Don't mimic color effects in black-and-white designs 263
Figure 9-6 A limited palette of colors 264

Chapter 10

Behaviors 267

- Figure 10-1** Different pointing devices 268
Figure 10-2 The insertion point and the pointer 269
Figure 10-3 A status indicator 271
Figure 10-4 Clicking a button 272
Figure 10-5 Double-clicking to select a word 272
Figure 10-6 Pressing a scroll arrow 273
Figure 10-7 Dragging to move an object 274
Figure 10-8 Using the Tab key to cycle through fields 276
Figure 10-9 Using the Return key to move the insertion point 277
Figure 10-10 A sample confirmation dialog box for the Escape key 278
Figure 10-11 Using Option-drag to make a copy of an object 279
Figure 10-12 Arrow keys 281
Figure 10-13 The function keys 284
Figure 10-14 Three ways of selecting information 287
Figure 10-15 Selection techniques 288
Figure 10-16 Expanding and shrinking a text selection 290
Figure 10-17 Extending text selections using the addition and fixed-point methods 290
Figure 10-18 Discontinuous selection within an array 292
Figure 10-19 Text selections 293
Figure 10-20 Selecting with Shift and arrow keys 296
Figure 10-21 Selecting with Option-Shift and arrow keys 296
Figure 10-22 Selection in an object-based graphics document 297
Figure 10-23 Selection in a bitmap-based graphics document 297
Figure 10-24 Field selection in an array 298
Figure 10-25 Column selection in an array 298
Figure 10-26 Range selection in an array 299
Figure 10-27 Discontinuous selection in an array 299
Figure 10-28 Intelligent cut and paste 302
- Table 10-1** Pointers 270
Table 10-2 How modifier keys change the movement of the insertion point with the arrow keys 283

Figure 11-1	Proper capitalization of screen elements	309
Figure 11-2	Clear button names	310
Figure 11-3	A poorly written alert box message	311
Figure 11-4	An improved alert box message	311
Figure 11-5	A well-written alert box message	311
Figure 11-6	Correct absence of a colon to introduce a list of options	312
Figure 11-7	Correct use of a colon	312
Figure 11-8	A help balloon	316
Figure 11-9	Help balloon for a button	319
Figure 11-10	Help balloon for a menu title	320
Figure 11-11	Help balloon for a menu item	320
Figure 11-12	Help balloon for a selected radio button	321
Figure 11-13	Help balloon for a checkbox	322
Figure 11-14	Help balloon for a group of radio buttons	323
Figure 11-15	Help balloon for a tool palette	323
Figure 11-16	Help balloons for an application icon and a document icon	324
Figure 11-17	Help balloon for a text entry box	325
Table 11-1	Translation chart for user documentation	307
Table 11-2	Categories of questions for help systems	315

About This Book

Macintosh Human Interface Guidelines describes the way to create products that optimize the interaction between people and Macintosh computers. It explains the whys and hows of the Macintosh interface in general terms and specific details.

Macintosh Human Interface Guidelines helps you link the philosophy behind the Macintosh interface to the actual implementation of interface elements. Examples from a wide range of Macintosh products show good human interface design, including individual and iterative examples. These examples are accompanied by descriptions and discussions of why to follow the guidelines. This book also contains examples of how *not* to do human interface design; they are marked as such and appear with a discussion that points out what's inappropriate and how to correct it.

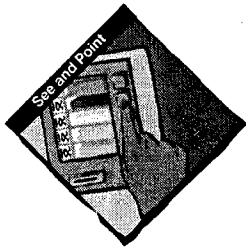
Who Should Read This Book

This book is written for people who design and develop products for use with Macintosh computers. If you are a designer, a human interface professional, or an engineer, this book contains information you can use to design and create products that fit the Macintosh model. It provides background information that can help you plan and make decisions about your product design.

Even if you don't design and develop products for the Macintosh, reading this book will help you to understand the Macintosh interface. This book is appropriate for managers and planners who are thinking about developing Macintosh products, as well as people who are interested in human interface design in general.

This book is written with the assumption that you are familiar with the concepts and terminology used with Macintosh computers and that you have used a Macintosh computer and some Macintosh applications.

Menus



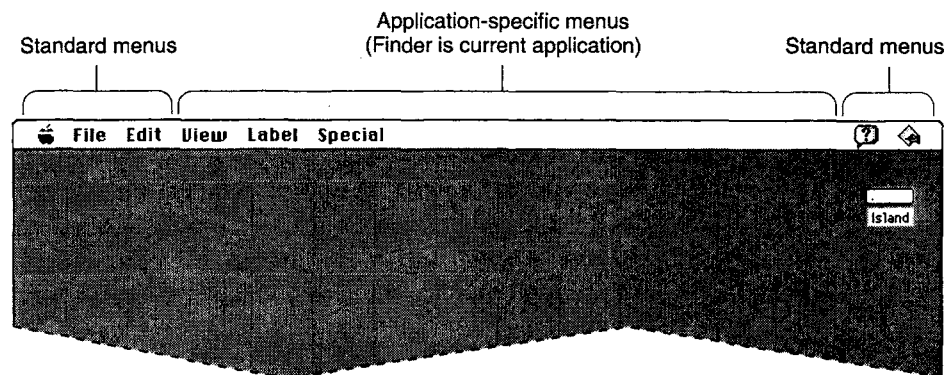
Menus are based on the principle of see-and-point. People don't have to remember command names because they can see all the options at any time and choose any available option. The menu bar also reflects the principles of perceived stability and aesthetic integrity. The menu bar provides a stable location for people to look for commands. It remains at the top of the screen no matter what the user is doing. To maintain the visual clarity of the interface, only the menu bar is visible until the user pulls down a menu to look at its contents. This feature allows people to concentrate on the main attraction, the content of the screen, while giving them easy access to the menus. Each application, each desk accessory, and the Finder have their own menu bars, containing standard menus and their own application-specific menus.

The Menu Bar

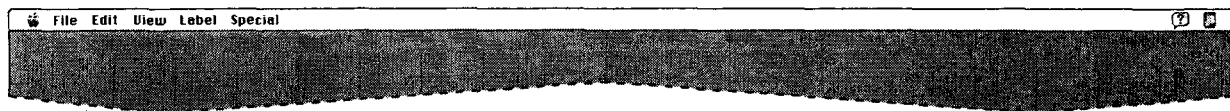
The **menu bar** extends across the top of the screen and contains words and icons that serve as the title of each menu. It should be visible and always available to use. Nothing should ever appear on top of the menu bar or obscure it from view. The menu bar should always contain the standard menus—the Apple menu, the File menu, the Edit menu, the Help menu, and the Application menu. The Keyboard menu is an optional standard menu that appears when the user installs a script system other than the Roman Script System. The titles of the standard menus never change. (The titles of the Apple, Help, Keyboard, and Application menus are icons rather than words.) The standard menus are described in the section “Standard Macintosh Menus,” beginning on page 98.

You can include as many menus in between these standard menus as are essential to your application and as fit on the smallest screen on which your application runs. It's a good idea to leave some room in the menu bar, both for localization and for menus added by third-party products such as utilities. Figure 4-2 shows the menu bar as it looks when the Finder is the current application, and points out the space you have for application-specific menus.

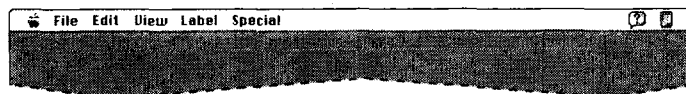
Menus

Figure 4-2 A menu bar

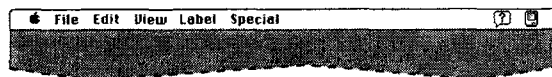
The width of the menu bar depends on the screen size of the user's monitor. Figure 4-3 shows menu bars of three different widths and how much space is used by the Finder's menu titles on each screen.

Figure 4-3 Three menu bars

19-inch screen



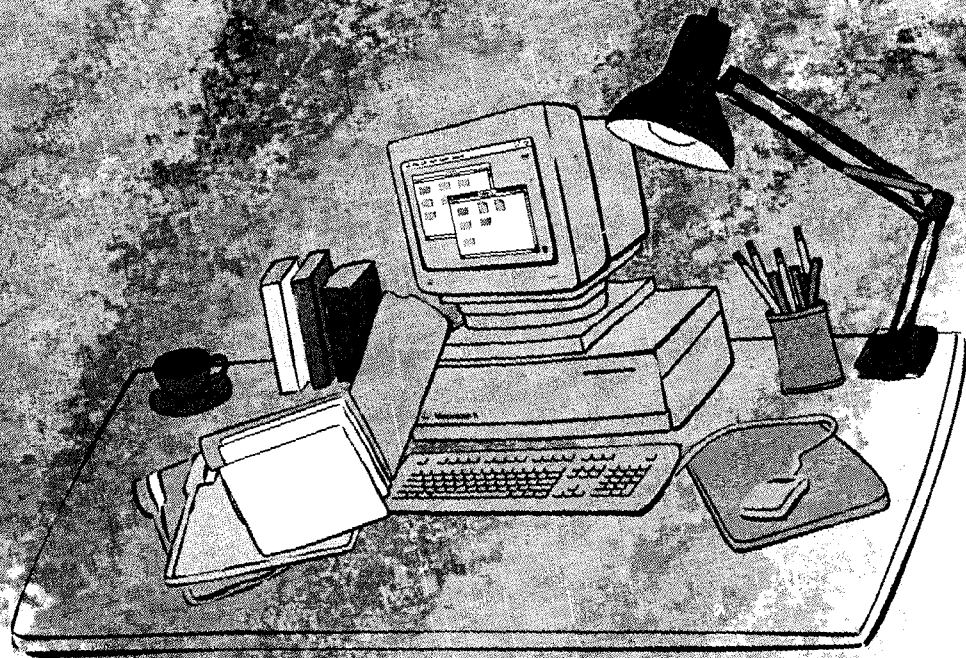
13-inch screen



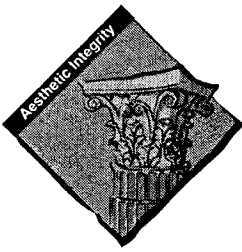
9-inch screen

Icons

8



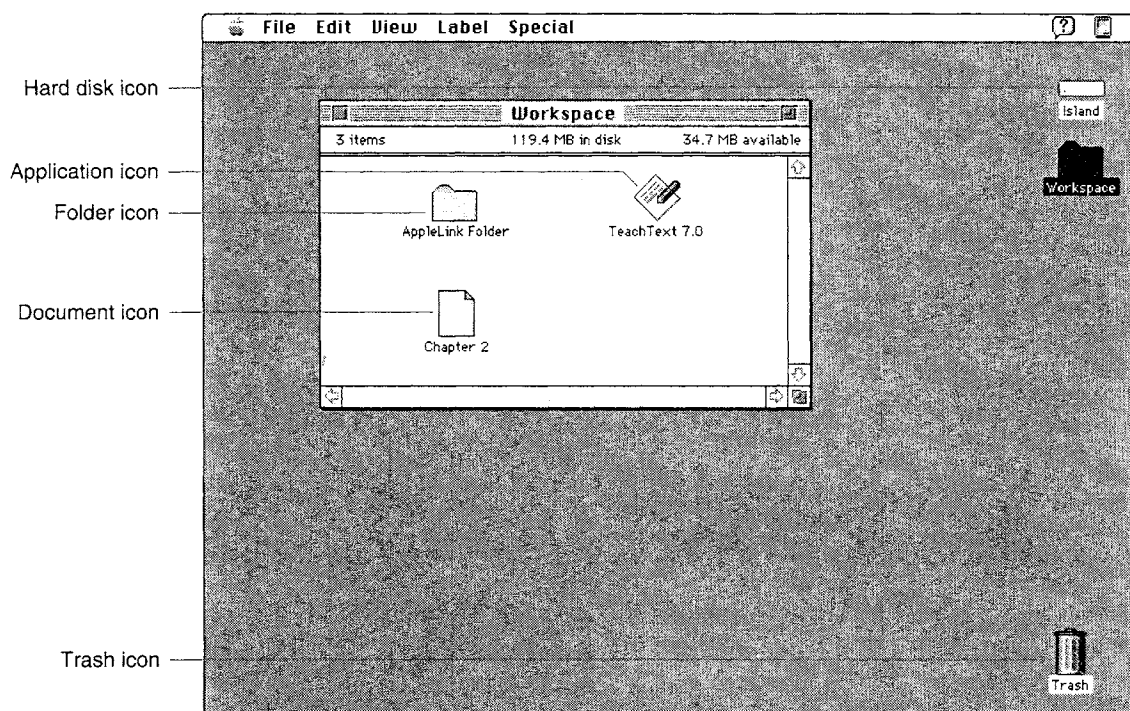
Icons



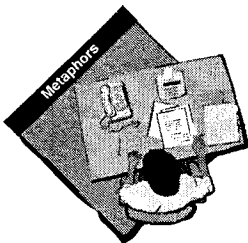
This chapter describes icons, their appearance, and their use in the Macintosh interface. It presents information on how to design icons and general guidelines for designing icons of different sizes and bit depths. This chapter also describes how to customize the standard icons you can provide for your products.

Icons are graphic representations of objects such as documents, storage media, folders, applications, and the Trash. Icons look like their real-world counterparts whenever possible. People can select, open, move, copy, and throw away icons. Figure 8-1 shows some icons displayed in the Finder.

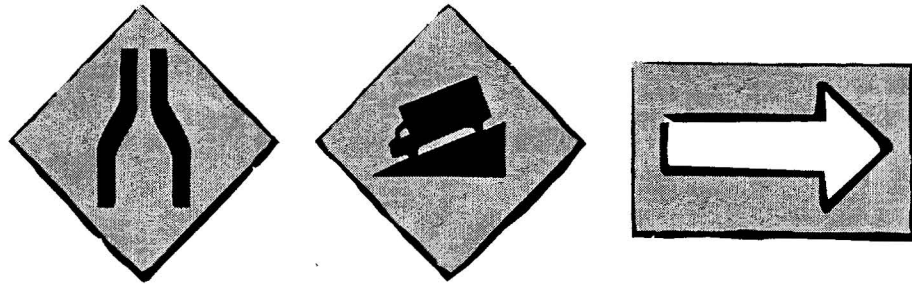
Figure 8-1 Common icons



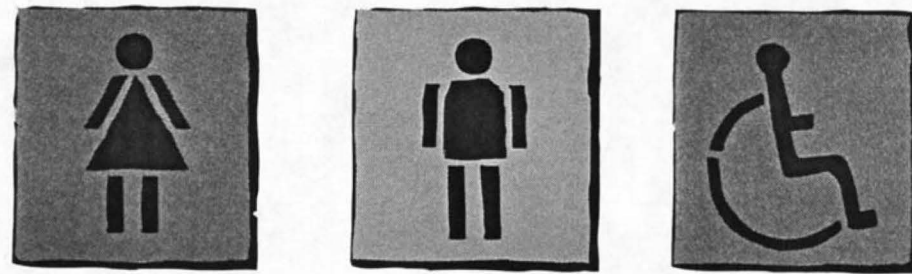
Why Icons Work



Icons work effectively in the Macintosh interface as representations of computer entities for several reasons. People often recognize pictures of things and understand them more quickly than they do verbal representations of the same things. For example, studies have shown that traffic signs that have symbols are more recognizable from a distance than signs that have only words. Figure 8-2 shows several examples of traffic symbols.

Figure 8-2 Examples of common traffic symbols

Symbols cross cultural and language barriers better than words do. Figure 8-3 shows examples of symbols that are used internationally. For example, at the Olympics pictures communicate ideas such as the locations of various events. People from around the globe must recognize the meanings of the signs in order to make their way to events, facilities, and services.

Figure 8-3 Examples of commonly-used international symbols

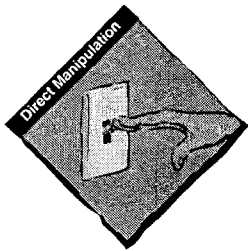
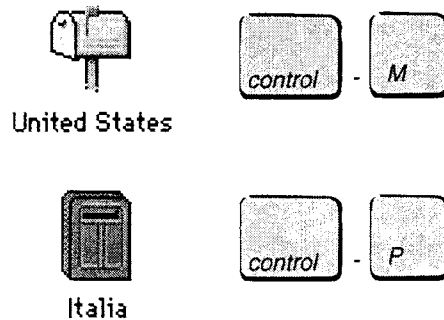
Symbols also take up less space than words that describe the same concept would. Imagine having to include the words written in many different languages on the signs. You can see that it's much easier in some cases to use graphics to represent concepts.

In the computer realm, it's generally easier to recognize symbols across systems than it is to remember keyboard commands. For example, a mailbox, even if it is of a different kind, is still recognizable as a mailbox. People identify a mailbox with sending and receiving communication. It's much harder for a user to remember that one system uses the command Control-M for mail, another system uses Control-P for post, and a third system uses Control-S for send.

Icons

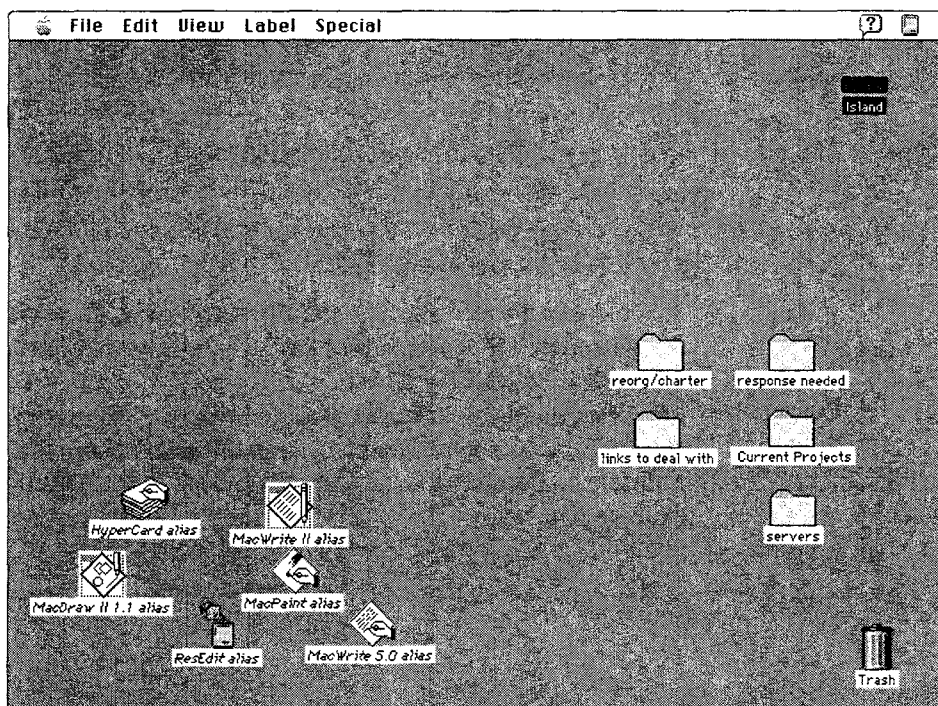
Figure 8-4 shows two examples of mailbox icons and their equivalent keyboard commands.

Figure 8-4 Symbols are easier to understand than keyboard commands



Icons provide direct access to items in the interface. People can see a folder, open it, and see its contents, or they can organize their desktop simply by grouping icons, rather than having to remember a lot of filenames and using lists to keep track of their files. Thus using icons contributes to the clarity and aesthetic integrity of the interface. Figure 8-5 shows a desktop with icon groupings.

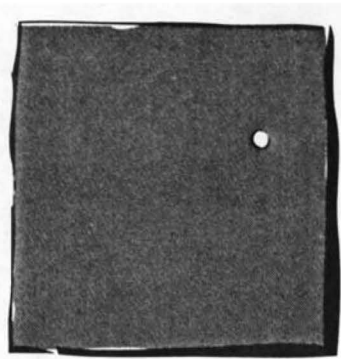
Figure 8-5 Grouping icons on the desktop



Limitations of Icons

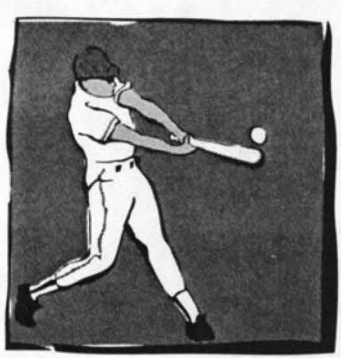
Designing the right icon that conveys your message to most people can be difficult. Sometimes it's difficult because icons need a context to provide successful communication. For example, what does the drawing shown in Figure 8-6 mean? It's a circle that could represent a wide variety of objects in the real world. From this figure, it's not clear what the image represents.

Figure 8-6 A confusing image



This image could represent a circle tool, a degree symbol, a ball, or a planet. It could be as many round things as people could imagine. When it appears in context with another object, the meaning instantly becomes clear, as Figure 8-7 shows.

Figure 8-7 Context clarifies the image

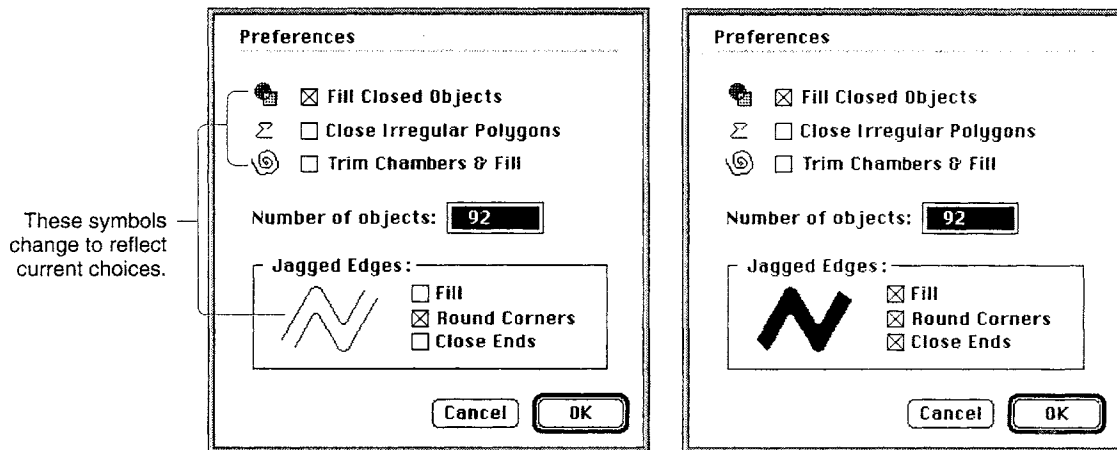


Icons

You can clearly see that in this context the circle is a baseball. This picture communicates an idea in a simple, graphic format. By adding the baseball player to the circle image, the context clarifies the meaning of the image.

In general, you can represent most nouns (people, places, and things) quite simply and easily in an icon. For example, you can draw a small picture that looks like a file folder to be a folder icon. Actions are much harder to portray in icons. How would you represent a save operation with an icon? You can overcome this difficulty by representing an action with an icon in combination with text. In fact, icons with label text are always more effective than either text or icons in isolation. Figure 8-8 shows a dialog box that uses icons that change to reflect the choices the user makes. The icons and text together help the icons give the user an idea of what to expect.

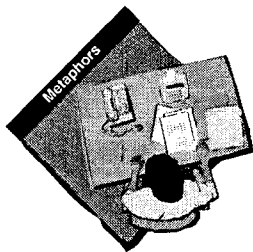
Figure 8-8 Icons with label text



Sometimes another interface element is more appropriate than an icon that isn't clear. For example, you could use text for error messages in dialog boxes more effectively than trying to communicate the same concept with symbols. Sometimes text is the simplest way to convey a concept, depending on the specific interface situation.

Designing Effective Icons

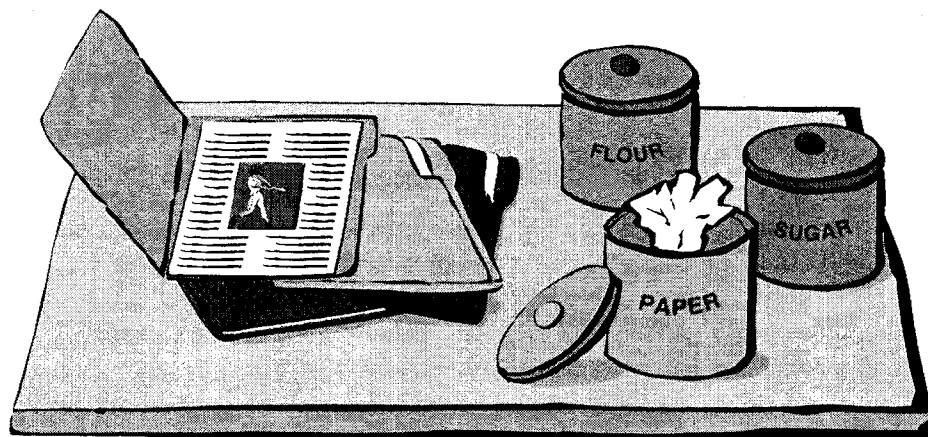
This section presents some basic guidelines for designing effective icons. Remember that all your icon designs must work in the Macintosh context. The desktop interface is based on an office metaphor using a desktop as the primary workspace. You want to build on this basis and diverge from it as little as possible. You must also consider other important facets of the Macintosh interface described here when you're designing icons. For information on designing color icons, see Chapter 9, "Color," beginning on page 257.



Use Appropriate Metaphors

You need to choose the appropriate metaphor to design effective icons. Folders are the appropriate metaphor for a storage container for documents because people use folders to store pieces of paper. In contrast, most people probably wouldn't use a kitchen canister for storing documents. People do use canisters to store things, and the shape of a hard disk platter is round. But it would require a major leap of logic for people to associate a canister as a storage medium for something like paper. Figure 8-9 illustrates this point.

Figure 8-9 A logical and an illogical metaphor



Icons



Think About Worldwide Compatibility

Your icon should be localizable for different regions around the world or should be designed with worldwide use in mind. For example, to localize an icon for receiving mail, you would substitute a post box for a mailbox in British system software. A worldwide icon is one that is understood universally. An example of an icon that is understood around the world is the document icon. Even though people in different locations around the world use different sizes of paper and different types of paper stock, they all still recognize the document icon as a representation of a document.

Figure 8-10 shows some examples of mailbox icons that have been localized for use in different countries.

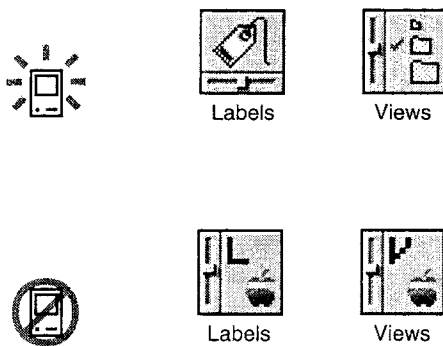
Figure 8-10 Localized mailbox icons



In general icons shouldn't be gratuitously cute. Humor typically doesn't translate well to other cultures or languages. Also, don't use inside jokes or pictures that represent code names for your icons. Although it might work to use such icons during your development process for product identification, be sure to remove them and replace them with appropriate icons *before* you ship. Symbols and colloquial language are usually culturally dependent, meaning that what one person relates to may have no meaning or may be an insult in another person's culture.

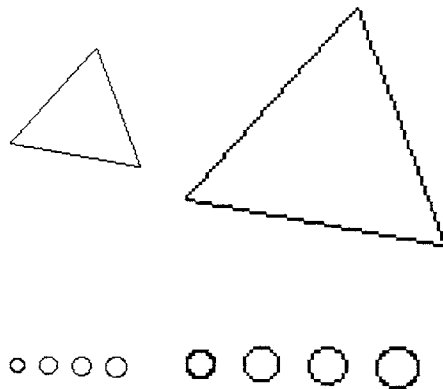
Avoid Text in Icons

Avoid using text in your icons whenever possible. Text in icons can be confusing, and it's not localizable to other regions, languages, or countries. It's appropriate to label icons to help the user recognize them and so that they can be read to a person with a visual disability. When you ship your product, every icon should have a name, but people can change the name of an icon at any time. Figure 8-11 shows an example of icons with text in them and icons that convey the idea much better without text.

Figure 8-11 Avoid text in icons

Design for the Macintosh Display

Your icons must look good at the current display resolution. You should use straight lines and 45-degree angles for the best appearance. Curves don't work well because they make the edges appear jagged. Figure 8-12 shows the jagged effects that curves and angles other than 45 degrees produce.

Figure 8-12 Certain shapes don't work well

Three-dimensional effects in icons are difficult to achieve in the Macintosh interface because they require shading and more angled lines. A large percentage of users have black-and-white monitors and thus complex shading may not display very well on their screens. If you decide to attempt to incorporate three-dimensional effects in your icons, make sure that a professional visual designer works on the design to ensure aesthetic integrity and compatibility with the Macintosh interface appearance.

Use a Consistent Light Source

On the Macintosh screen the light source always comes from the upper-left corner of the screen. Therefore icons and other elements have drop shadows on the lower-right side. Use the light source consistently, so that shading is consistent throughout the interface. Figure 8-13 shows some desktop entities that have drop shadows consistent with a light source at the upper-left corner of the screen.

Figure 8-13 A consistent light source

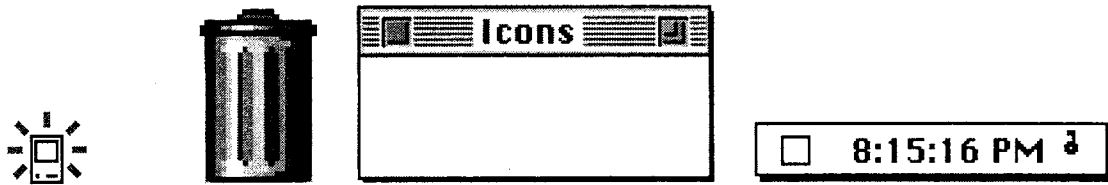
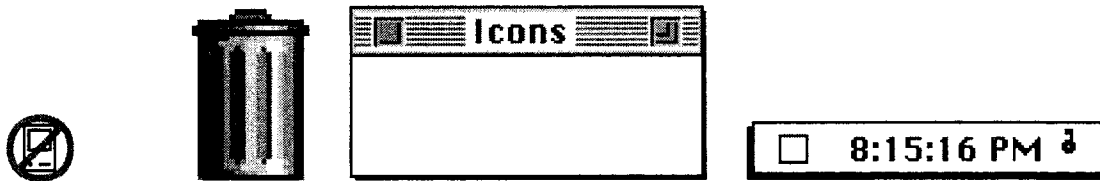


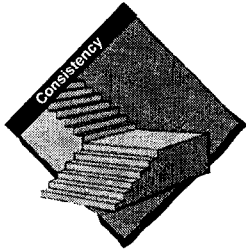
Figure 8-14 shows some desktop images that have different light sources and inconsistent drop shadows.

Figure 8-14 Inconsistent light sources



Optimize for Your Target Display

You should optimize your design for the display on which it will most often be seen. For example, some games run on only 4-bit and black-and-white monitors. In this case, you should optimize your design by choosing colors from the 4-bit color palette, rather than starting with the 8-bit version and trying to scale down from there.

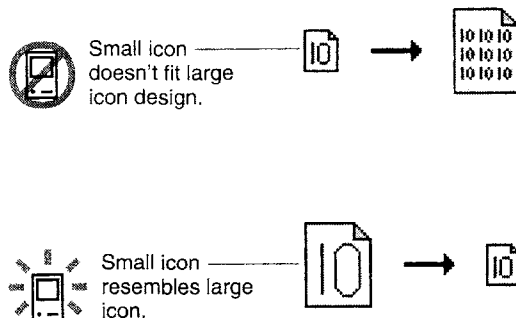


Maintain a Consistent Visual Appearance in an Icon Family

Maintain a close visual relationship between all members of an icon family. Color versions of icons should resemble the black-and-white versions. Users should be able to easily recognize standard interface elements and icons across all monitor types. Users can have several monitors connected to a computer and several computers on which they use your applications. Your application icon should look consistent when a user changes the bit depth of a monitor or moves your icon from a color monitor to a black-and-white monitor.

Design the large (32-by-32 pixel) icon first, and then adapt the design to the small icon. You can leave out inessential details in the small version of your icon, but it shouldn't look significantly different from the large version. Figure 8-15 shows how starting with the design of the large icon and adapting it for the small icon works better than starting with the design of the small icon.

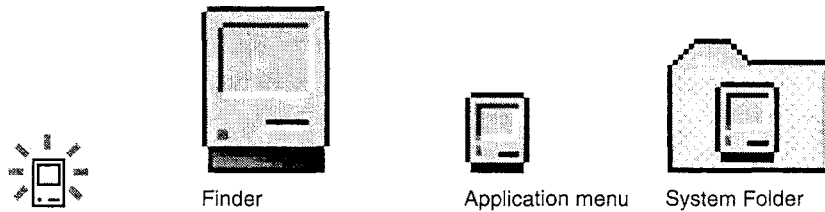
Figure 8-15 Design the large icon first and base the small icon design on it



Use Icon Elements Consistently

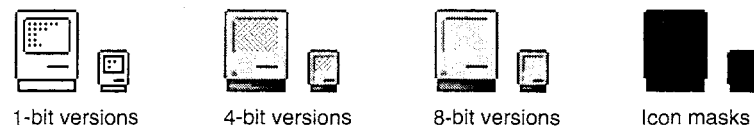
Use icon elements consistently throughout your designs. If there is an existing shape for an icon element, don't change it. For example, don't invent new designs for entities that have a standard design in system-provided icons, such as folders and documents. For example, in Figure 8-16, the icon of the Macintosh computer is the same when it appears in the Finder, in the Application menu, and within the System Folder icon. Unless you are representing a different model of Macintosh computer, use the Macintosh Classic icon to represent the Macintosh computer. People often assume that different shapes have different meanings and may try to read meaning in where none is intended. Users can learn what icon elements represent when they are used consistently.

Icons

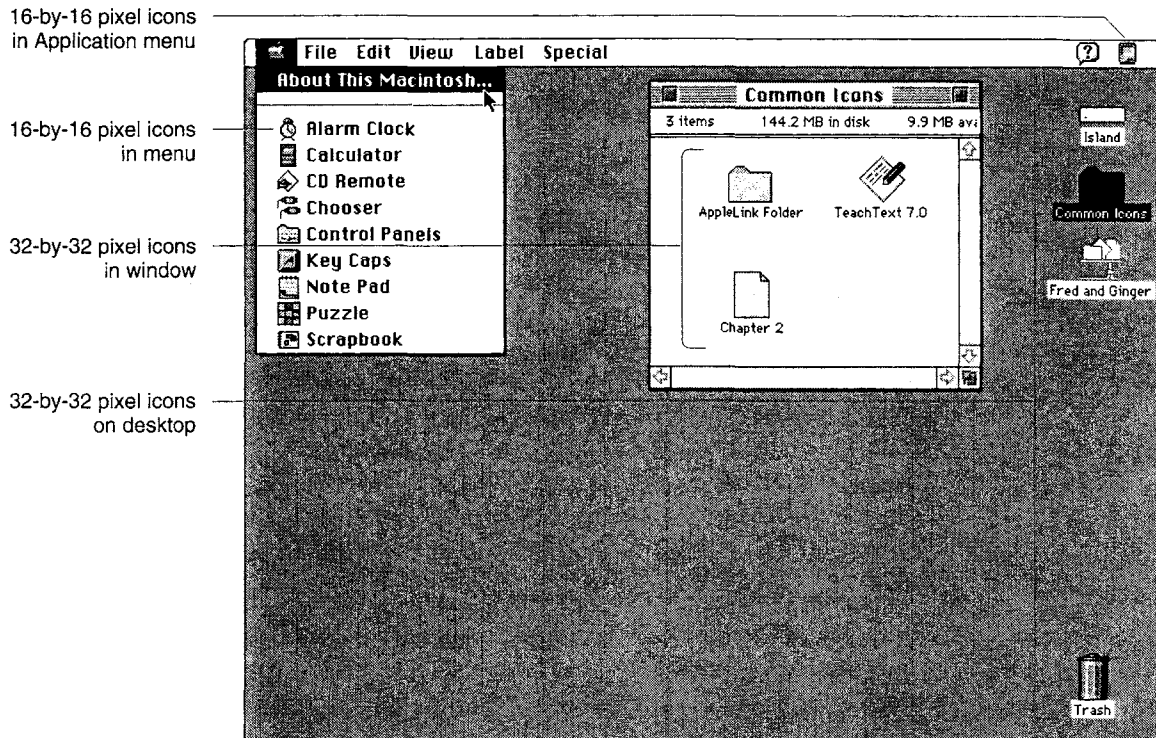
Figure 8-16 Consistent use of icon elements

The Finder Icon Family

For display in the Finder, for each icon you should provide an entire icon family, which consists of large (32-by-32 pixel) and small (16-by-16 pixel) icons, each available in three different versions of color: black and white (1-bit color), 4-bit color, and 8-bit color. You also need to provide an icon mask for each size of icon. The Finder uses the icon mask to cut out space on the desktop for the display of the icon. Figure 8-17 shows a family of icons for System 7.

Figure 8-17 An icon family

The 32-by-32 pixel icons appear on the desktop, and, if the user chooses “by Icon” from the View menu, these icons also appear in Finder windows. The 16-by-16 pixel icon appears as the Application menu’s title when your application is active. It also appears next to your application’s name in the Application menu and in Finder windows when the user chooses “by Small Icon” from the View menu. Figure 8-18 shows examples of two sizes of icons and some instances of where they appear in the Macintosh interface.

Figure 8-18 Different sizes of icons

You can also provide icons for other entities that your application creates, such as customized document icons, preferences file icons, and stationery icons. The entire group of icons that you distribute with your product is known as a suite of icons. A suite of icons includes the families of icons for each type of icon that you distribute.

A monitor displays the highest-quality icon that its screen allows. Table 8-1 shows which icons are displayed on monitors with different display capabilities based on the icon families that you provide.

Table 8-1 Icon display on monitors of different bit depths

Icon set provided	What 8-bit color monitor displays	What 4-bit color monitor displays	What black-and-white monitor displays
8-bit, 4-bit, black-and-white icons	8-bit icon	4-bit icon	Black-and-white icon
8-bit, black-and-white icons	8-bit icon	Black-and-white icon	Black-and-white icon
4-bit, black-and-white icons	4-bit icon	4-bit icon	Black-and-white icon
Black-and-white icon	Black-and-white icon	Black-and-white icon	Black-and-white icon

Icons

See the chapter “Finder Interface” in *Inside Macintosh: Macintosh Toolbox Essentials* for information about the icons you provide and how to create a bundle resource for your application.

An Icon Design Process

This section presents an icon design process used at Apple. Of course, there are many ways to complete a task, and you may work in a way that doesn't lend itself to using this process. However, you should read this section to find out important principles that contribute to successful and effective icons.

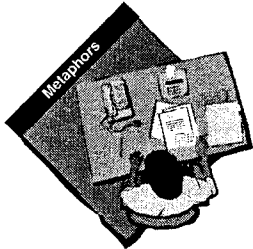
You can't design one icon in your icon family in isolation. All the icons in a family—large, small, and different color depths—are incarnations of the same icon, so the basic design must work for all of them. Be flexible in adapting your design to all bit depths and sizes. Icon design is an iterative process. During the design process, you may need to redesign one version of an icon when you find it doesn't translate well to another version.

If an icon represents hardware that users are familiar with, such as a printer or a disk drive, the icon should resemble the hardware as closely as possible. If the icon represents a desktop entity, such as a document or a folder, it should resemble its physical world counterpart. If you need to design an icon for a more conceptual entity, such as a network or some kind of memory, you can use one of the following approaches. Making the icon representative of the function of the software is a good approach. If the function is complex and hard for novice users to understand, think about how you could explain the idea to someone who doesn't use a computer and try to generate some images that way. Often the terms you use and the analogies you come up with to explain the concept can provide clues for visual images.

You can also make an icon representative of a product name. This may work for your product in one location, but remember that some product names, and thus product icons, are often not localizable. For example, in the United States, an icon for extensions could have something to do with an extension cord. In other languages, the word used for extension cords may have nothing to do with extensions, and therefore an icon based on the word *extension cord* would be meaningless. Another drawback to this approach is that product names are often not finalized until late in the development process, so you might not have much time in which to design an icon based on the final product name.

Icons

A final approach to designing conceptual icons is making the icon look like the window that results from opening the icon. For example, if a window that appears when the icon is opened is very distinct in appearance from other windows, you can make an icon look like that window. You need to be very careful that all your icons do not look like miniature windows.



It is often easiest to create icons that represent objects (nouns) rather than actions (verbs). For example, the function of deleting a document is represented by a trash can (an object) rather than by some image of the action of deleting. Thinking of an object that is representative of the function of your icon is the key to good conceptual design. Remember that for every image you generate, you need to consider the advantages and disadvantages of the idea in regard to your audience before deciding on the final design.

Here's an outline of the suggested steps in an icon design process. The sections that follow go into the details of some icon design guidelines.

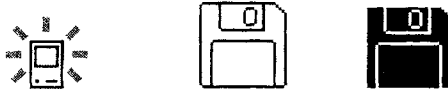
1. Start by designing the black-and-white version of the large (32-by-32 pixel) icon. Follow the guidelines given in the next section, "Black-and-White Icons."
2. Colorize the black-and-white version of your icon in 8-bit color. Use colors from the Apple icon colors palette (see the section "The Apple Icon Color Set" on page 240). For more information about creating color versions of black-and-white icons, see "Design for Black and White First" on page 263 in Chapter 9, "Color."
3. Translate the 8-bit version of your icon into 4-bit color. If you can't find appropriate colors in the 4-bit palette, try going back to step 2 and swap colors in the 8-bit version. You can use a dithered pattern of two colors to create the illusion of a new color for the 4-bit icon.
4. Create the mask.
5. Create matching small versions (16-by-16 pixel) of the black-and-white, 8-bit, and 4-bit icons.
6. Look at the icon family on different desktop backgrounds and with several effects such as selection and labeling to make sure that the icons look good.
7. Do some usability testing to make sure that your target audience understands your icons, doesn't confuse them with other icons, and identifies them as a family of icons.

ResEdit is a useful tool for accomplishing steps 3, 4, and 6.

Black-and-White Icons

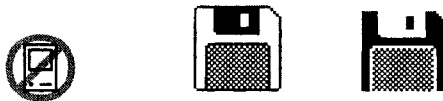
You should begin by designing a black-and-white icon. In general, you should use an outline of one black pixel to create the icon border. Use a minimal number of black pixels in the icon so that the icon's appearance is noticeably different when selected. The Finder automatically inverts the black pixels and the white pixels when the user selects the icon to create the selected appearance. Figure 8-19 shows an example of a well-designed icon that changes significantly when selected.

Figure 8-19 A well-designed icon and its selected version



If you use too much black or 50 percent gray in your icon, the icon doesn't appear significantly different when the pixels are reversed for selection. Figure 8-20 shows an example of an icon with too much black and 50 percent gray.

Figure 8-20 A poorly designed icon and its selected version



Color Icons

Macintosh system software ships with full-color icons that appear on color monitors. Your application can also provide color icons.

Icons

Don't design a color icon that's substantially different from your black-and-white icon. When you add color to an icon, it's best to leave the one-pixel black outline and other black lines that form the icon, and fill the icon in with color. Coloring or graying the icon's outline makes the icon appear less distinct on the desktop. Remember that the user can change the background color of the desktop as well as its pattern, so your icon may not be displayed against the background on which you designed it. If you use ResEdit version 2.1 or later to create your icons, it provides a way to look at your icon against different backgrounds to see whether your design is effective in various environments such as black-and-white displays or color displays of different bit depths. Figure 8-21 shows how icons with a black border look on a gray background.

Figure 8-21 Icons with a black outline

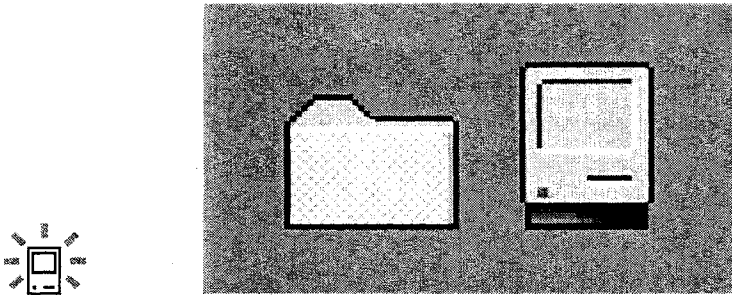
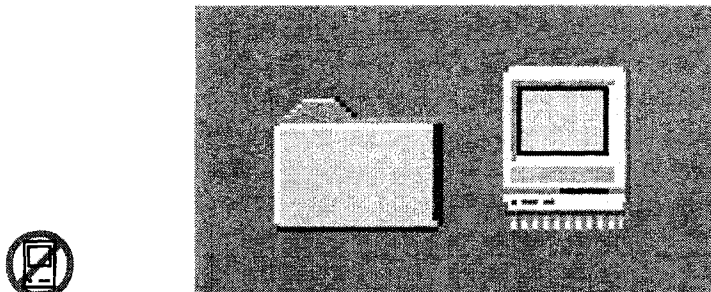


Figure 8-22 demonstrates how an icon appears less distinct from its background without the strong black border.

Figure 8-22 Icons without a black outline



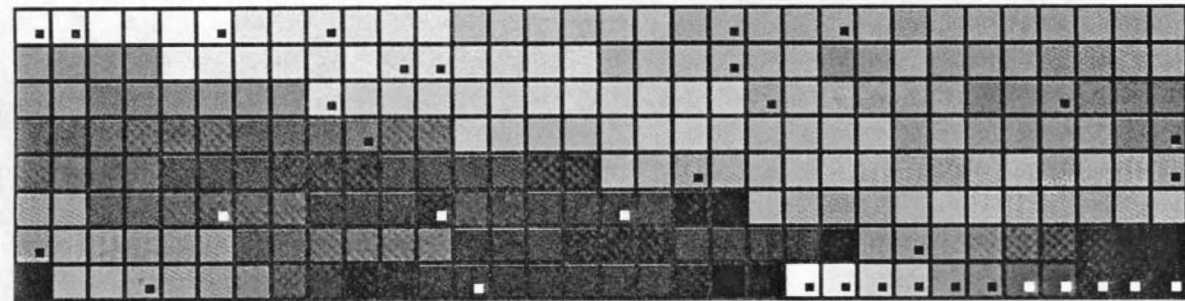
Icon Colors

This section describes the colors and color techniques that you should use when you design your color icons.

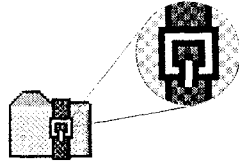
The Apple Icon Color Set

Figure 8-23 shows a palette of the standard 256 colors with a mark on each of the 34 colors used for icon design in system software. If you use ResEdit version 2.1 or later to design and create your icons, the Finder icon family editor provides easy access to these colors. Choose Apple Icon Colors from the Color menu. This command sets the palette in the editor (which is similar to the palette in most graphics applications) to contain the 34 colors used for Finder icons. See *ResEdit Reference* for information on using ResEdit.

Figure 8-23 Standard 256-color palette with icon colors marked



This entire set of 34 colors was chosen to be subtle. Subdued colors avoid a “circus” effect on the screen. If you use too many of the same types of colors, people can’t discern what is important as easily. Ramps of color based on the initial colors chosen were created to provide shading and blending capabilities. Some light colors were included in the set to be used for large areas. The colors from the Apple logo were included because those colors have a strong Apple identification. The Apple logo colors are very bright and should be used sparingly. The total set of colors provides maximum flexibility in design. You can combine these colors in a dithered pattern to provide additional color effects. Figure 8-24 shows an example of creating a dithered color to use in your icons.

Figure 8-24 An example of dithered color in an icon

The icon colors were chosen for icon design in system software. It was necessary to limit the number of colors in the set to create consistency across all Apple icons. There are at least 120 system icons. If any number of the 256 system palette colors were used in each icon, the total effect would be a disparate appearance rather than a coherent look to all system icons. Using only 34 colors makes the system icons look like they belong together. In a design scheme that is so large, fewer colors look better.

Degradation of the Color Set Across Monitors

If the default color table colors aren't available, the system software gracefully degrades to black and white. First the operating system tries to match 8-bit colors. If it can't successfully match the colors you specify with those in the system palette, then it displays the 4-bit icon that you supply. If the operating system can't find or match the 4-bit colors, then it displays the icon in black and white. The system software won't substitute colors that aren't visually close to colors that you assigned. If you choose colors other than the 34 marked in Figure 8-23, use them for detail and not for essential parts of your windows or icons.

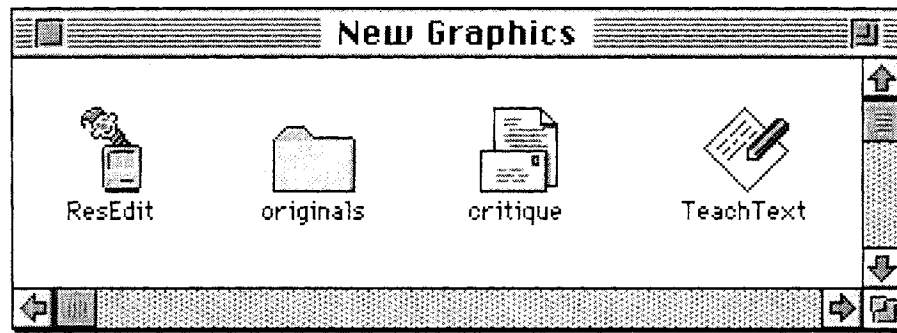
Selection Mechanism for Color Icons

When a color icon is selected, the color decreases in brightness. This means that the colors appear darker when selected. On a color monitor, a black-and-white icon turns gray when selected. On a black-and-white monitor, a black-and-white icon uses reverse video to show selection. To make selected items appear distinct from unselected ones, use light colors for large areas. Note that only the 34 colors shown in Figure 8-23 get darkened when a color icon is selected. This means that if you use colors other than those 34 colors in large amounts in an icon, that icon will not get darkened and therefore will not look selected.

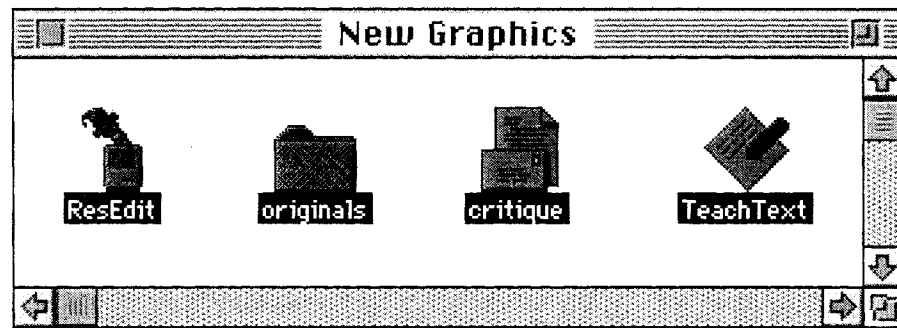
Icons

Figure 8-25 shows a set of control panel icons as they appear on the desktop and the same icons in their selected states.

Figure 8-25 Color icons and their selected states



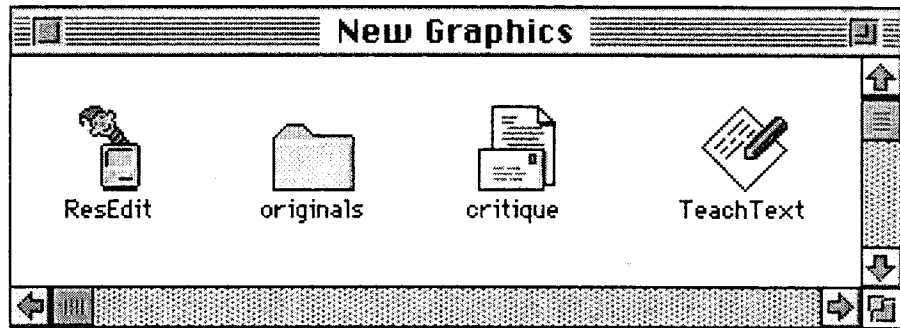
Color icons



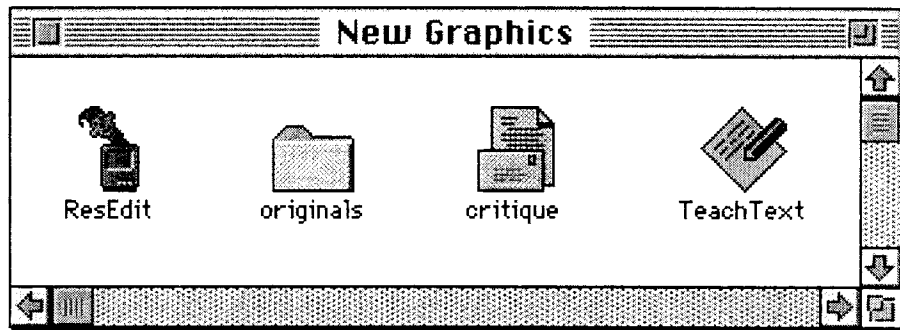
Selected color icons

Color Labeling Mechanism for Color Icons

The labeling mechanism tints color icons toward the label color chosen by the user from the Label menu. Figure 8-26 shows some icons and the same icons in their labeled state. To provide system support for this technique, it was necessary to limit the number of colors used in icons. Using the 34 identified icon colors does not guarantee that labeled icons will look good, only that they will actually look labeled. As with icons that are selected, the tinting applies to only the 34 colors. As a result, if you use colors other than the 34 colors in your icon, the icon will not look labeled when a label color is applied.

Figure 8-26 Color icons and their color-labeled states

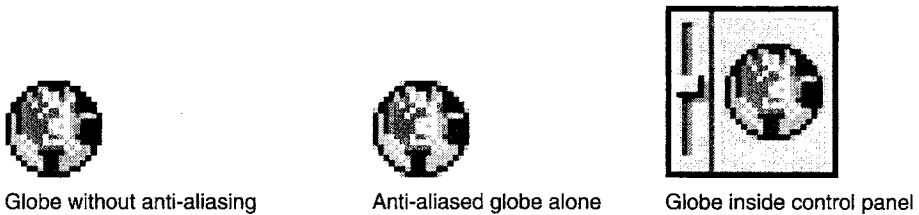
Color icons



Labeled color icons

Anti-Aliasing

A technique for enhancing the appearance of your icons is to smooth angular or curved lines by coloring pixels on jagged edges. This technique is called *anti-aliasing*. Change the pixel color where you can see a visual break in the outline of a black-and-white icon. Figure 8-27 shows an icon before anti-aliasing, after anti-aliasing, and then in the context of a control panel icon.

Figure 8-27 Correct anti-aliasing

Icons

In anti-aliasing, you typically add pixels to an outline shape. Since the Finder uses only one mask for each size in the icon family, make sure that all your icons have the same outline shape. That is, when you anti-alias icons, don't add pixels or shadows to the outline shape of color icons. Figure 8-27 shows how anti-aliasing works well within an icon. The Finder uses the icon mask for alignment and transformation effects, so make sure that the mask and all your icons are appropriate for each other.

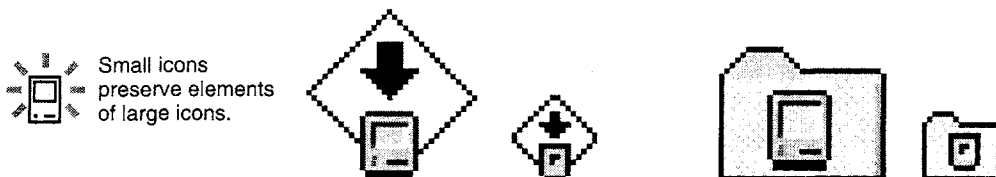
If you add too much anti-aliasing to the icons, they appear smooth, but also more fuzzy. While some people prefer this appearance, the Macintosh desktop appearance relies on crisp-looking icons. Gray outlines create a fuzzy image on the desktop. If people perceive something fuzzy on their screen, they may assume that something is wrong with their eyes or their display. Avoid creating this appearance if at all possible.

Small Icons

If you do not provide a 16-by-16 pixel icon, the Finder reduces the 32-by-32 pixel icon based on an algorithmic formula. The algorithm simply shrinks the icon and typically creates black areas, creating less pleasing visual results. If you provide a 16-by-16 pixel icon, however, you can optimize its design by removing pixels when necessary.

When you design a small version of your 32-by-32 pixel icon, preserve as many graphical elements of the icon as possible. In essence you want to provide the same icon in a smaller size. Typically you have to remove some pixels to reduce the visual clutter. However, don't eliminate significant elements, or the smaller version of the icon may look different from the larger version. Figure 8-28 shows icons that a designer carefully scaled and tuned to preserve key elements of the icons' designs.

Figure 8-28 Consistently designed small icons

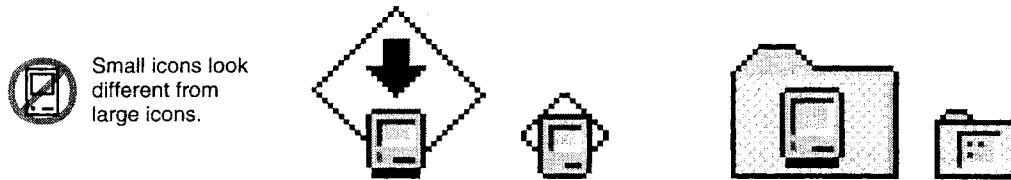


Icons

After you've created the small icon, verify the accuracy and clarity of the small icon by trying to design a large icon based on its design. If the large icon you end up with based on your small icon is different from the original large icon, something is not working about your small icon. In this case, you should consider redesigning the small icon, making sure to incorporate the key graphical elements of the original large icon.

In Figure 8-29 the small icons don't match their corresponding 32-by-32 pixel versions. If you have difficulty distinguishing the consistencies or inconsistencies, it's a good idea to consult with someone who specializes in graphic design to design or review your icons.

Figure 8-29 Inconsistently designed small icons



Default and Custom Icons

You can provide custom icons for your application and its associated documents and files. If you don't provide custom icons, the Finder displays default icons in most cases. There are no default icons provided for preferences files or control panels, so you must provide icons for these.

Since there are so many icons that users see and deal with, it's important to create consistent sets of icons. Users should be able to easily identify and locate the icons they're looking for, which is more possible if related icons look related. Each icon family consists of the same icon in two sizes and three bit depths. Each type of icon family needs to look like a class of objects. For example, all document-related icons should look like documents. The entire group of icons that belongs to your product, called a suite of icons, should have a common appearance that identifies all the icons as being related to your product.

Icons

Application Icons

Because applications are usually used to create documents, the application icon uses a tilted document page to represent the documents the application creates. In this way, a relationship is established. The hand is also part of the default application icon. Figure 8-30 shows the default icon that appears in the Finder if you don't supply a custom icon for your application.

Figure 8-30 Default application icons



Although it's best to be consistent, if you must design an icon different from the default shape, be sure to use one of the standard application icon elements, either the page or the hand, in your icon. Figure 8-31 shows two custom application icons that use both the tilted document page and the hand elements.

Figure 8-31 Custom application icons



You can customize the application icon by adding graphics to it. Use graphics that convey meaning about what your application does. If you can't think of elements that represent the overall function of your application, it's OK to use your company identification. However, using your logo is limiting because your company will probably make more than one application in its lifetime and you will want a distinct icon for each one.

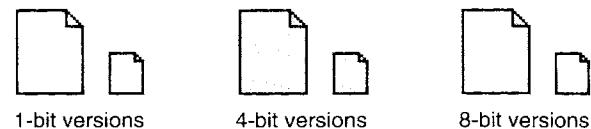
Don't design an application icon that is completely unrelated to the basic shape of the application icon. People won't get the benefit of visual clues that help them identify your icon as an application. Figure 8-32 shows two examples of application icons that totally violate the guidelines presented here. They don't give the user much information about what they do, although they might be fun to look at.

Icons

Figure 8-32 Examples of bad application icons

Document Icons

Documents are the files in which users store the content they create in an application. The document icon uses the outline of a page with a turned-down upper-right corner. This shape evokes the concept that a document is like a piece of paper, a typical storage medium for information in an office. Figure 8-33 shows the default document icons that appear in the Finder if you don't supply a custom icon for your documents.

Figure 8-33 Default document icons

You can customize this document page icon so that it relates to your application icon by adding the same graphics you use in your application icon. Or you can add graphics to the document page that indicate the content that your documents hold. Figure 8-34 shows the relationship between an application icon and its document icon.

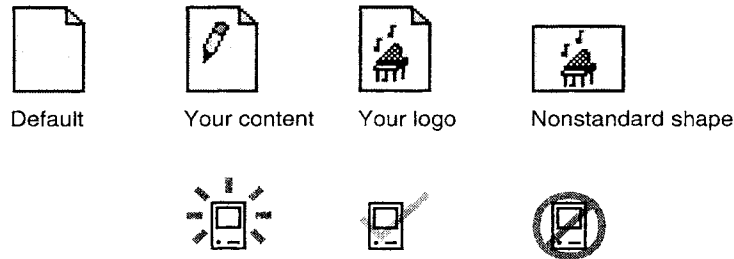
Figure 8-34 Application icon and document icon with the same graphic element

As with the application icon, it's OK to use your company identification in a document icon. Be sure not to change the shape of the document icon unless the document that your application creates is fundamentally different from most documents. One example of a document that *is* fundamentally different is a stack. The stack icon is a metaphor that represents a stack of cards, similar to a stack of index cards that you might use to store notes or related information.

Icons

Figure 8-35 shows some examples of custom document icons that show the range of acceptable and unacceptable customizations.

Figure 8-35 Acceptable and unacceptable custom document icons



Several conventions exist for conveying document types. You can use these visual clues or others that have meaning for your documents' contents. Documents that are text-only or primarily text use broken lines on the document page. Page layout documents typically include a filled rectangle and broken lines to indicate that the document contains text and pictures. Graphics document icons use geometric shapes or other graphics or graphics tools such as paintbrushes. The standard representation for a file of type 'PICT' is a circle, a square, and a triangle. Use this symbol to indicate files of this type. Figure 8-36 shows document icons with all of these symbols.

Figure 8-36 Document icons with standard symbols



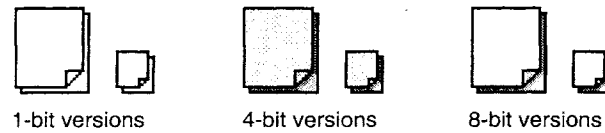
Stationery Pad Icons

A stationery pad is a template with standard contents that the user can create from a document. Each time a user opens a stationery pad, a new document is opened with the same contents as the stationery pad. The stationery pad icon uses the outline of a page, similar to the document icon, but with the lower-right corner turned up and a second page visible in the background.

Icons

Figure 8-37 shows the default icons that appear in the Finder if you don't supply a custom icon for stationery pads.

Figure 8-37 Default stationery pad icons



You can customize a stationery pad icon by adding graphic elements to the stationery document page. Your stationery pad icon should look just like your document icon except that it has the second page in the background and the lower-right corner of the first page turned up.

Query Document Icons

Query documents are documents that contain instructions used to get data out of or into a database. The query document icon uses the outline of a document page with a database element attached to its lower right. Figure 8-38 shows the default icons that appear in the Finder if you don't supply a custom icon.

Figure 8-38 Default query document icons



You can customize a query document icon by adding graphics to the document page. Be sure to maintain the outline of the icon and the volume symbol that represents the database. See *Inside Macintosh: Interapplication Communication* for information on using the Data Access Manager.

Icons

Edition Icons

An edition is a file that is created when a user chooses Create Publisher from the Edit menu. The edition icon has a rectangular outline in a horizontal orientation. Figure 8-39 shows the default icons that appear in the Finder if you don't supply a custom icon for editions.

Figure 8-39 Default edition icons



You can customize an edition icon by putting a different graphic inside the rectangle. Maintain the horizontal orientation and the double-dotted line of the icon that identify it as an edition icon. See *Inside Macintosh: Interapplication Communication* for information on the Edition Manager human interface and implementing the Edition Manager.

Preferences Icons

Many applications have preferences files that store information about a user's preferences. If you supply this file or create one after the user stores settings for your application or documents, you'll need to create an icon for it. Preferences file icons often contain document icons with radio buttons, as shown in Figure 8-40.

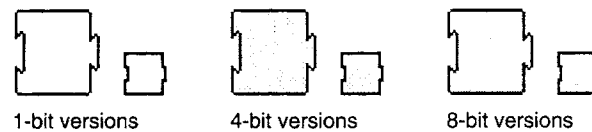
Figure 8-40 Preferences file icons



Extension Icons

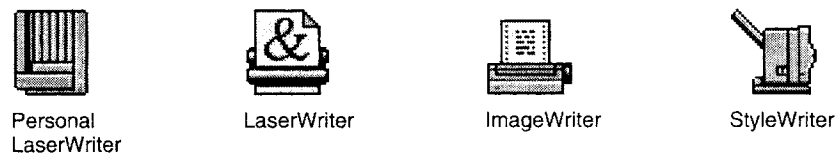
Extensions are software that add a feature or capability to the operating system. An extension icon looks like a puzzle piece. Figure 8-41 shows the default icons that appear in the Finder if you don't supply a custom icon for your extension.

Icons

Figure 8-41 Default extension icons

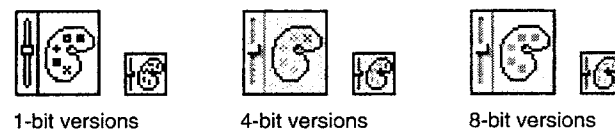
You can customize an extension icon by adding a graphic to the puzzle piece. You can display the puzzle piece in a horizontal or a vertical orientation with the protruding part facing any direction.

One exception to the standard form for extension icons are icons that represent Chooser extensions. Chooser extensions appear in the Chooser and provide people with a visual idea of the service they are choosing. Chooser extension icons should look as much as possible like the devices they represent. Some examples of Chooser icons are shown in Figure 8-42.

Figure 8-42 Examples of Chooser icons

Control Panel Icons

The control panel icon is a square with a slider on it to identify it. The slider also appears on the Control Panels folder. You can add a graphic to the square to customize the icon. You can display the slider in either a horizontal or a vertical orientation. Figure 8-43 shows the icon family for the Color control panel.

Figure 8-43 Icons for the Color control panel

Icons

Movable Resource Icons

The Finder displays default icons for fonts, keyboard layouts, and sounds, also known as movable resources. The icon looks like a document icon reversed in orientation. The user installs these resources by dragging the icon to the System Folder icon. The user can remove a movable resource by opening the System file icon (or Font Folder icon) in the System Folder and dragging its icon out of the System file. TrueType fonts have a character in three sizes on the icon, whereas bitmapped fonts only have a single character. When a user opens a font icon, the Finder displays a window with a sample of the font in it. Figure 8-44 shows some font icons.

Figure 8-44 Font icons



Bitmapped font



TrueType font

Sound icons have a speaker symbol on them, as shown in Figure 8-45. When a user opens the sound icon, the sound plays.

Figure 8-45 A sound icon



Sound

Keyboard Icons

A keyboard icon represents a localized keyboard layout or input method. If you develop keyboards, input methods, script systems, or keyboard resources, you need to provide icons. (Keyboard icons appear in the Keyboard menu and in the Keyboard control panel.) You need to create a 16-by-16 pixel icon in 1-bit and 4-bit color. If you don't create a keyboard icon, then the system provides a default icon. Figure 8-46 shows the default keyboard layout icon and the default input method icon.

Icons

Figure 8-46 The default keyboard layout and input method icons

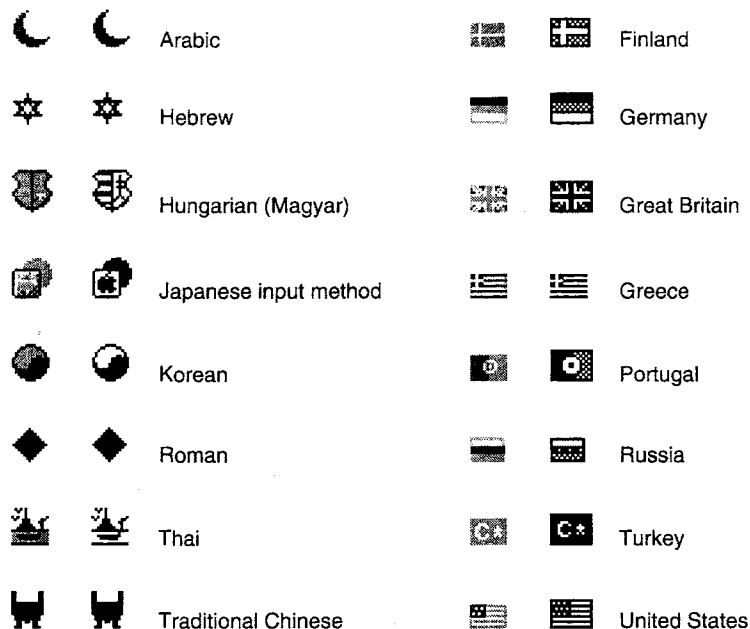
Default keyboard layout icon



Default input method icon

Keyboard icons are a special case of icons. Since they represent a specific type of software they follow design rules different from those for other icons users see on the desktop. The guidelines presented in this section apply only to keyboard icons.

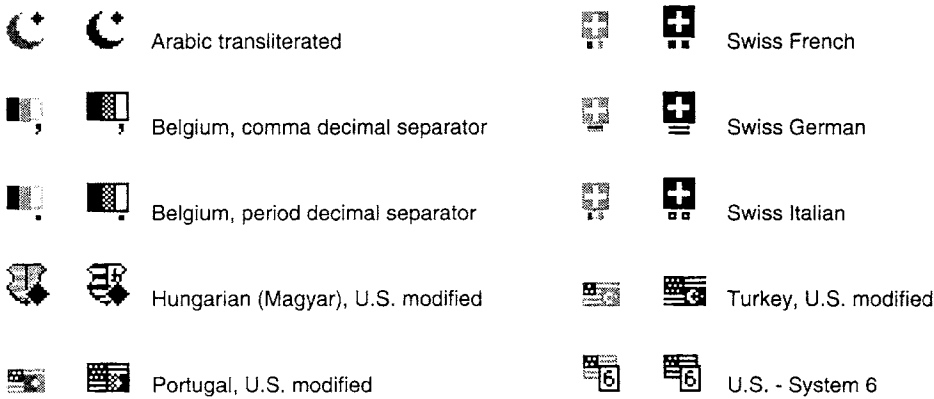
For a keyboard icon, use a solid symbol to represent a keyboard layout for a region that is larger or smaller than an area that can be represented by the flag of a country or province. For example, a diamond represents the Roman Script System, which is used in the United States, Central America, South America, Australia, New Zealand, and most of Europe. Use the flag of a country or province if the keyboard layout is used primarily in that area. For example, the Union Jack represents the keyboard layout localized for use in the United Kingdom. Be sure to use the colors that appear on the nation's flag. Figure 8-47 shows some keyboard icons for script systems and localized keyboard layouts.

Figure 8-47 Examples of keyboard icons

Icons

You can also add a visual indicator to the keyboard icon to show some modification. Use a superscript diamond to indicate a QWERTY transliteration, which is a mapping of sounds from a language to the Roman keyboard layout. Figure 8-48 shows some flag symbols with additional indicators.

Figure 8-48 Examples of modification indicators on keyboard icons



When you design the black-and-white version of a flag icon, use black and a 50 percent gray pattern. These choices provide the best contrast and legibility. To avoid confusion between flags of similar design, use the pattern substitutions for colors shown in Table 8-2.

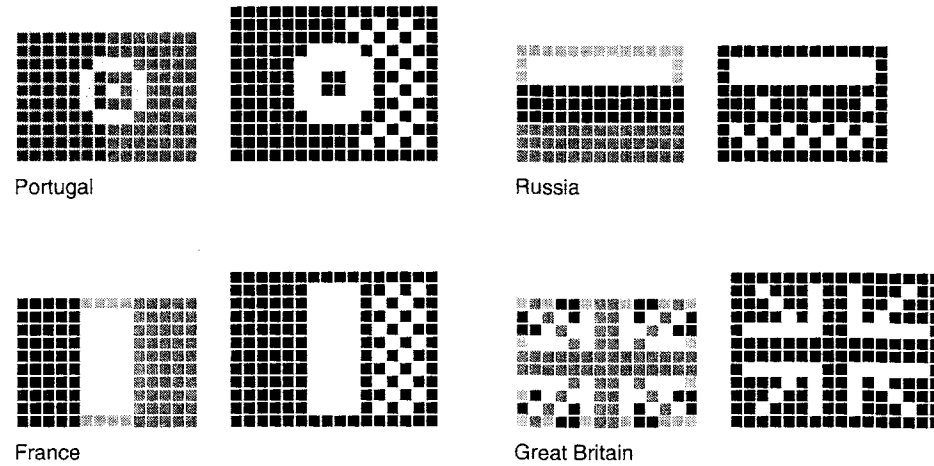
Table 8-2 Pattern substitutions for colors in keyboard icons

Color	Pattern
  Black or blue	 Black
 Red	 50 percent gray
 Light blue	 25 percent gray
 Green	 Diagonal stripes
  White or yellow	 White

Icons

Figure 8-49 shows some keyboard icons that use the correct pattern substitutions.

Figure 8-49 Enlarged keyboard icons with correct color substitutions



See the section “The Keyboard Menu” beginning on page 125 in Chapter 4, “Menus,” and *Inside Macintosh: Text* for information on the Keyboard menu. See the chapter “Finder Interface” in *Inside Macintosh: Macintosh Toolbox Essentials* for more information about displaying custom icons. That chapter also provides information on how to use the bundle resource to associate these icons with your application.



> \$29.95 USA
> \$38.95 CANADA

Macintosh Human Interface Guidelines

by Apple Computer, Inc.

Macintosh Human Interface Guidelines provides authoritative information on the theory behind the Macintosh "look and feel" and the practice of using individual interface components. This book includes many examples of good design and explains why one implementation is superior to another. Anyone designing or creating a product for Macintosh computers needs to understand the information in this book.

Read this book to learn

- how people interact with computers
- the principles of the Macintosh human interface
- interface behaviors, such as responses to user input from the mouse or keyboard
- interface elements, such as menus, windows, dialog boxes, controls, and icons
- the importance of designing for worldwide markets and for people with disabilities
- effective use of color
- guidelines for using language clearly and consistently
- suggestions for creating an effective design process, including how to manage complexity in software and how to extend the interface beyond these guidelines
- sources of further information

At the back of the book is a checklist for evaluating how well your program conforms to the human interface guidelines.

To get the most out of this book, you should be familiar with Macintosh computers. If you are a programmer, you should also consult *Inside Macintosh: Macintosh Toolbox Essentials* for complete information on the technical implementation of your program and its interface.

Apple Computer, Inc.
20525 Mariani Avenue
Cupertino, CA 95014
408-996-1010
TLX 171-576

Addison-Wesley Publishing Company

