

THE BMUG GUIDE TO RESEdit

EXPANDED FOURTH EDITION

ZEN
and the ART of RESOURCE EDITING

“Hands down,
the best way to
learn ResEdit!”

— BOB LEVITUS,
MAC GURU & RACONTEUR

Customize your
Macintosh®
with ResEdit

DERRICK SCHNEIDER
AND HANS HANSEN

Hayden
Books



ZEN

and THE ART of RESOURCE EDITING





Photographs of the Great Buddha at Kamakura
by Hans Hansen. Japan, 1993.

ZEN

and The ART of RESOURCE EDITING

The BMUG Guide to ResEdit

EXPANDED FOURTH EDITION

Written and Revised By
Derrick Schneider
Hans Hansen



A BMUG Press Publication



Published by Hayden Books

Zen and the Art of Resource Editing: The BMUG Guide to ResEdit, Expanded Fourth Edition

©1995 Derrick Schneider and Hans Hansen

All rights reserved. Printed in the United States of America. No part of this book may be used or reproduced in any form or by any means, or stored in a database or retrieval system, without prior written permission of the publisher except in the case of brief quotations embodied in critical articles and reviews. Making copies of any part of this book for any purpose other than your own personal use is a violation of United States copyright laws. For information, address Hayden Books, 201 W. 103rd Street, Indianapolis, Indiana 46290.

Library of Congress Catalog Number: 95-80296
ISBN: 1-56830-244-4

This book is sold as is, without warranty of any kind, either express or implied. While every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information or instructions contained herein. It is further stated that the publisher and author are not responsible for any damage or loss to your data or your equipment that results directly or indirectly from your use of this book.

97 96 95 4 3 2 1

Interpretation of the printing code: the rightmost double-digit number is the year of the book's printing; the rightmost single-digit number is the number of the book's printing. For example, a printing code of 95-1 shows that the first printing of the book occurred in 1995.

Trademark Acknowledgments: All products mentioned in this book are either trademarks of the companies referenced in this book, registered trademarks of the companies referenced in this book, or neither. We strongly advise that you investigate a particular product's name thoroughly before you use the name as your own.

Apple, Mac, and Macintosh are registered trademarks of Apple Computer, Inc.

Limits of Liability and Disclaimer of Warranty

BMUG's licensor(s) makes no warranties, express or implied, including without limitation the implied warranties of merchantability and fitness for a particular purpose, regarding the software. BMUG's licensor(s) does not warrant, guarantee, or make any representations regarding the use or the results of the use of the software in terms of its correctness, accuracy, reliability, currentness, or otherwise. The entire risk as the results and performance of the software is assumed by you. The exclusion of implied warranties is not permitted by some jurisdictions. The above exclusion may not apply to you.

In no event will BMUG's licensor(s), and their directors, officers, employees or agents (collectively BMUG's licensor) be liable to you for any consequential, incidental or indirect damages (including damages for loss of business profits, business interruption, loss of business information, and the like) arising out of the use or inability to use the software even if BMUG's licensor has been advised of the possibility of such damages. Because some jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitations may not apply to you. BMUG's licensor's liability to you for actual damages from any cause whatsoever, and regardless of the form of the action (whether in contract, tort (including negligence), product liability or otherwise), will be limited to \$50.



Special Thanks

Noah Potkin, who helped make the previous editions of this book shine. Much of the Zenness of this book's style is due to his enlightened influence.

Thanks, Noah—may you enjoy croissants and espresso wherever you find yourself.

Dedication

To **Randy Simon** and **Steve Costa**, who encouraged us to make a book on ResEdit. Without both of you this book would not have existed.

And to all the people who continue to be enthusiastic about the Zen book after so many years.



The Hayden Books Team

Don Fowley

PUBLISHER

Karen Whitehouse

ACQUISITIONS MANAGER

Pat Gibbons

MANAGER OF PRODUCT DEVELOPMENT

Brad Miser

DEVELOPMENT EDITOR

Steve Mulder

COPY & PRODUCTION EDITOR

Ray Robinson

MARKETING MANAGER

Rosemary Lewis

PUBLISHING COORDINATOR

Kelly Dobbs

PRODUCTION MANAGER

Laurie Casey

PRODUCTION TEAM SUPERVISOR

Paul Gilchrist

MANUFACTURING COORDINATOR

**Heather Butler, Dan Caparo, Kim Cofer, Tricia Flodder, Aleata Howard,
Erika Millen, Beth Rago, Erich J. Richter, Karen Walsh, Christine Tyner, Robert Wolf**

PRODUCTION

Jeanne Clark

INDEXER

Hans Hansen

COVER & INTERIOR DESIGNER

Composed in the Adobe Myriad and Adobe Minion

multiple master typefaces.

Hayden Books
201 West 103rd Street
Indianapolis, IN 46290
(800) 428-5331 voice
(800) 448-3804 fax

America Online: Hayden Bks
AppleLink: hayden.books
CompuServe: 76350,3014
Internet: hayden@hayden.com

Visit the Hayden Books site on the World Wide Web at:

<http://www.mcp.com/hayden>

About BMUG

BMUG is a membership-based nonprofit organization dedicated to helping users of graphical interface computers. It represents the interests of over 12,000 Macintosh users in more than 50 countries.

BMUG started as a small user group in 1984, shortly after the introduction of the Macintosh. As a nonprofit corporation, BMUG strives to give the plain, unbiased truth about product performance and the industry in general. We don't sell advertising in our newsletters, and we make it quite clear that we will not exchange good reviews for product donations. BMUG is neither affiliated with nor receives monetary support from Apple Computer or any other for-profit entity.

This book was made possible by the knowledge and support of the people within BMUG. If you have any questions or comments about this book, please feel free to contact us. If you are interested in becoming a member or finding other books created by BMUG, contact us for further information.



BMUG, Inc.

1442A Walnut Street, #62
Berkeley, CA 94709

voice: (510) 549-2684

fax: (510) 849-9026

email: bmug_info@bmug.org

web: <http://www.bmug.org/>



About the Authors

Derrick Schneider got lost on his way to veterinary school, and ended up as a computer programmer at Bank of America, a lonely representative of the long-haired Berkeley liberals he went to college with. He attributes this detour to BMUG, where he discovered a deep-set fascination with computers and a willingness to take on all kinds of large projects. As a result, he coauthored *Zen and the Art of Resource Editing* and authored *The Tao of AppleScript*, and scripted the production of BMUG's 700-page disk catalog. It shouldn't be too surprising that part of the reason for the detour from vet school was a GPA that wouldn't get him admitted to any self-respecting campus, though amazingly enough he did manage to get a diploma, unlike most BMUG zealots.

Hans Hansen has always had a strong need to create things. Whether it was pioneering his high school yearbook's use of computers for production or developing BMUG as a publisher of quality Macintosh products, he has been addicted to using the Mac to go where no desktop publisher has gone before. During his nine years at BMUG, he has had the pleasure of working with many talented authors, including coauthoring BMUG's first book, *Zen and the Art of Resource Editing*, and developing its best-selling *The Tao of AppleScript*. Hans also developed and organized the production of BMUG's 700-page disk catalog that won acclaim because of its original use of Apple event scripting. Spending the last four years as BMUG's Publications Manager and Editor-in-Chief of *The BMUG Newsletter*, he has followed the Macintosh industry all over the world. Hans recently left BMUG to pursue his lifelong dream of founding his own business as an independent production company specializing in the creative composition of books, CD-ROMs, and Web sites.



Acknowledgments

We would like to thank the following people for their contributions:

Brad Miser for helping us through our eastern philosophy series of books.

Steve Mulder for keeping everything consistent and looking good.

Pat Gibbons and **Karen Whitehouse** for believing in our vision of a fourth edition.

All the original authors who shared their knowledge of resources: **Jens Peter Alfke**, **Tom Chavez**, **Chris Holmes**, **Lisa Lee**, **Brendan McCarthy**, **Leonard Morgenstern**, **Brian Novack**, **Blaise R. Pabón**, **Rick Reynolds**, **James W. Walker**, and **Steve Yaste**.

Gray Shaw, **Tim Holmes**, **Robert Lettieri**, **Wendy Masri**, **Tim Bodine**, and **Raines Cohen** for their work on previous editions.

Cliff Stoll for his enthusiasm and advice.

David Rogelberg for his support through the years.

Robert Lettieri for his great suggestions and influence on the design of this book.

Jeff Benjamin for getting us a copy of Guide Composer on last-minute notice so we could give you the Zen ResEdit Apple Guide on the CD.

Ted Nace and all our friends at Peachpit Press for their support and hard work.

Wendy Masri for the handmade brush stroke title artwork.

Utopia Grokware for the inspirational screen shot of Flowfazer.

Chuck Carter for his unique rendition of the ResEdit clown.

BMUG (past and present) with the help of **Kelly**, **Michele**, **Alisa**, **Sean**, **Lisa**, **Robert G.**, **Michael**, **Art**, **Stan**, **Gerald**, **Steve**, **Jeff**, **Randy**, **David S.**, **Robert L.**, **Judi**, **Herb**, **Greg**, **David M.**, **Brian**, **Fred**, **Peter**, **Rob**, **Ann**, **Karla**, **Bruce**, **Jeannene**, and **Scott R.**

Our friends and family who put up with our long nights and weekend absences: **Harper**, **Mario**, **Chris**, **Tim**, **Mitch**, **Jim**, **John**, and **Tigran**.

*Special thanks to **Eiji** and **Kayin**, who kept our breaks interesting and allowed us to vent our frustrations.*



Contents at a Glance

Foreword	<i>by Cliff Stoll</i>	xvi
Preface	<i>by Derrick and Hans</i>	xvii
Introduction	What Is ResEdit, and Why Zen?	xx
CD Introduction	About the Zen CD-ROM	xxiii
Chapter 0	Jumping Ahead	1
Chapter 1	What Are Resources?	11
Chapter 2	Working with Resources	25
Chapter 3	Deciphering Icons	47
Chapter 4	Recognizing Patterns	71
Chapter 5	Correcting Cursors	83
Chapter 6	Styling Text	97
Chapter 7	Font Types	105
Chapter 8	Tuning Keyboards	125
Chapter 9	Selecting Menus	143
Chapter 10	Dialog Boxing.....	157
Chapter 11	Color Palettes	175
Chapter 12	Application Resources	189
Chapter 13	Finder Resources	201
Chapter 14	System Resources	225
Chapter 15	ResEdit Resources	235
Appendix A	Guide to Four-Letter Words	249
Appendix B	Meet Your Creator	257
Appendix C	Hex Editing	261
Appendix D	Exploring the Zen CD-ROM	265
Glossary		271
About BMUG		285
Index		291

Table of Contents

0	Jumping Ahead	1
	Installing ResEdit	2
	Backing Up Your System	3
	Mystery Spot One	4
	Mystery Spot Two	5
	Mystery Spot Three	6
	Mystery Spot Four	6
	A Mystery No More	7
	Resurrecting Your System	8
1	What Are Resources?	11
	How Do They Work?	14
	Looking at a Program's Resources	15
	Resources Are the Foundation of the Mac	21
	Resources Overview	22
	Safety Tips—What Not to Do!	23
	The Buddha's Guide to ResEdit Enlightenment	23
2	Working with Resources	25
	Using Resources in Files	30
	Moving Resources	31
	Types and Creators	32
	Giant Sucking Sounds	33
	Other Resources to Move Around	36
	PICT Resources	37
	FKEYs	39
	One Resource, One ID	41
	Working with Resources Overview	42
	Safety Tips—What Not to Do!	45
	The Buddha's Guide to ResEdit Enlightenment	45
3	Deciphering Icons	47
	Dialog Icons	48
	cicn Icons	53
	Icon Masks	56

Finder Icons	60
Icon Families	61
Changing the Folder Icon	63
Small Icons (the Other Kind)	65
Icons Overview	66
The Buddha's Guide to ResEdit Enlightenment	68
4 Recognizing Patterns	71
'PAT ' Resources	72
ppat Resources: The Secret Behind the	
Color Desktop	74
But I Want Bigger Patterns!	76
The PAT# and ppt# Resources	78
ResEdit Patterns	79
Patterns Overview	79
The Buddha's Guide to ResEdit Enlightenment	81
5 Correcting Cursors	83
The CURS Resource	84
The Hot Spot	85
Cursor Masks	87
Color Cursors?	88
Animated Cursors	89
The acur Editor	89
Adding to an Animation	90
You and Your Arrow	91
Cursor Utilities	92
Cursors Overview	93
The Buddha's Guide to ResEdit Enlightenment	95
6 Styling Text	97
'STR ' Resources	98
STR# Resources	99
TEXT and Styl Resources	101
Text Resources Overview	102
Safety Tips—What Not to Do!	102
The Buddha's Guide to ResEdit Enlightenment	103
7 Font Types	105
FONT Resources	107
FOND Resources	107
NFNT Resources	109

sfnt Resources	110
What's in a Font	111
Editing Font Resources	112
The FONT/NFNT Editor	112
The FOND Editor	114
Editing sfnts...Not!	117
Renaming or Renumbering a FOND	118
Inside PostScript Font Files	119
Those Weird Filenames	120
What's Inside	120
Uses for Bitmap Fonts	121
Font Overview	122
The Buddha's Guide to ResEdit Enlightenment	123

8 Tuning Keyboards 125

Behind the Scenes	127
The KCHR Editor	128
Remapping Ordinary Keys	129
Remapping Modifier Keys	131
The KCHR Editor Menus	132
Dead Keys	134
KCHRs in International Systems	136
Small Icons with KCHRs	137
Keyboard Layouts in the Finder	138
Keyboards Overview	139
Safety Tips—What Not to Do!	140
The Buddha's Guide to ResEdit Enlightenment	141

9 Selecting Menus 143

Picking a MENU	145
The MENU Editor	146
Except... ..	152
Menus Overview	153
Safety Tips—What Not to Do!	154
The Buddha's Guide to ResEdit Enlightenment	155

10 Dialog Boxing 157

Where Do Dialogs Come From?	158
Editing the Dialog Items	160
DITL Menu Items	164

The DLOG and ALRT Resources	167
Numbering	169
Color Dialogs	171
Dialog Boxes Overview	172
Safety Tips—What Not to Do!	173
The Buddha's Guide to ResEdit Enlightenment	173
11 Color Palettes	175
Color Lookup Tables	176
The clut Resource Editor	178
Color Palette Cycling	181
What's a pltt Resource?	185
Color Palettes Overview	186
Safety Tips—What Not to Do!	187
The Buddha's Guide to ResEdit Enlightenment	187
12 Application Resources	189
The vers Resource	190
BNDLs and FREFs	193
Bundles of AppleScript	195
Bundles of Stationery	196
Application Resources Overview	197
Safety Tips—What Not to Do!	199
The Buddha's Guide to ResEdit Enlightenment	199
13 Finder Resources	201
More Memory for the Finder	202
Is Your Application "Busy or Missing"?	204
When Is a Menu Not a Menu?	207
Quitting the Finder	209
Clippings	211
Changing the Find File Progress Bar	213
Using the Fonts Folder for More than Fonts	214
More Interesting Strings in the Finder	216
Disabling the ZoomRects	218
Finder Resources Overview	218
Safety Tips—What Not to Do!	221
The Buddha's Guide to ResEdit Enlightenment	222

14 System Resources	225
The mcky Resource	226
The Secret About Box	228
Changing the Default Creator of Screen Shots	229
Welcome to Macintosh	230
System Resources Overview	232
Safety Tips—What Not to Do!	233
The Buddha's Guide to ResEdit Enlightenment	234
15 ResEdit Resources	235
Installing Editors	236
Using Templates	238
Editing the RMAP	240
Verify Resource	243
Setting Up Your Own Icons	244
ResEdit Resources Overview	245
The Buddha's Guide to ResEdit Enlightenment	248
A Guide to Four-Letter Words	249
B Meet Your Creator	257
C Hex Editing	261
Real Uses of the Hex Editor	263
D Exploring the Zen CD-ROM	265
ResEdit	266
Resources	266
Icons, Icons, and More Icons	267
Mounds of Patterns!	267
Tons of Cursors! No Waiting!	267
And That's Not All!	268
Cool Utilities	268
ResEdit Editors and Templates	269
Authors' Highlights	270
Glossary	271
About BMUG	285
Index	291

Zen and the Art of Resource Editing

When I got hold of the first edition of this book, I hadn't played with ResEdit in ages. I had some old version which I rarely used, and I had little idea of what I could do with it. I had never taken the time to sit down and learn it, because there were no instructions.

I was in the Bay Area for a while, so I stopped in at a BMUG meeting and picked up a copy of the book. I took it home that night and began to play with it. Gadzooks! I had never known that any of this could be possible! Reading through this book, I spent that entire night modifying my menus, changing my trash, and editing dialogs to say all sorts of properly cool things.

The really wild thing about ResEdit is that you can use it on any Mac program. Just zap your resources here and there, and all of a sudden your Mac is really your Mac—an extension of your personality. Because everything about the Mac—the sounds, the fonts, and a zillion other things—lives within a resource, ResEdit can tailor those resources to meet your needs. Of course, you need to tell it what to do.

However, this is not a book to read casually. You won't learn anything by curling up with a homespun quilt next to a roaring fire and reading through this book. You've got to abandon the family for a day or so, roll up your sleeves, lock yourself in your computer room, and get your keyboard dirty. Sit down with ResEdit, which comes with the book, turn to the first chapter, and just start hacking away. The chapters will tell you how to do the basic stuff, and then will give you a few ideas about other possible uses. However, there's nothing stopping you from going off and trying your own wild ideas or exploring. Poke around a little—you'll learn oodles about the way your Mac works. When you finally come out of your locked room into the light of day, you'll have a working knowledge of ResEdit that you can apply for as long as you've got your Mac.

— *Cliff Stoll*

It's been more than five (!) years since the idea of doing a BMUG ResEdit book was suggested to Derrick by one of BMUG's regulars. The idea stuck, and the book was soon underway. Derrick enlisted a variety of people from around BMUG to write sections on specific resource types. Hans and Noah joined in transforming it from a disorganized hodgepodge of files into a real book. Through the various problems we encountered, ranging from an early lack of authors to budgetary and political concerns, we were able to rush out the book to meet our deadline of the January 1991 Macworld Expo. Our goals for sales at Expo were conservative; 400 books sold would cover costs, and BMUG would have gotten its money back from the venture. Our actual results were surprising: close to 1,000 *Zen* books were sold in those four days. This success was even more surprising when we compared our thin book to Addison-Wesley's huge *ResEdit Complete* being sold around the corner. It was then that we realized that not only did ResEdit and the book have an audience, but BMUG had one too. The book was so successful that we immediately began planning the second edition.

For the second edition, each of us had goals for how the book would look and read. Derrick was intent on cramming in as much stuff as possible, including information on the as-yet-unreleased System 7 for which ResEdit 2.1 was designed. Hans and Noah wanted to completely redesign the problematic layout and cover from scratch. We all wanted to surpass the first edition, and we did.

The second edition was more than twice the size, but more important, it was easier to read and more organized. It focused more on the practical aspects of ResEdit, including some neat things you could do with the just-released System 7. This book, too, surprised us with its success, selling about 3,500 copies with little or no advertising out of the BMUG office and at the 1991 Boston Macworld Expo. While we were contemplating what we would add to the third printing, Peachpit Press became interested in distributing the book.

Zen and the Art of Resource Editing

Well, those two elements combined nicely to produce the third edition, our first mass-market release of the book. Not only did this book do very well, it was translated into Japanese and seen in bookstores around the world (while we were all in Paris for an Apple Expo, we came across it in a window display).

As sales of the third edition were winding down, Hans came up with the idea of doing a CD-ROM full of resources, utilities, and other cool ResEdit stuff. This would provide people with a huge set of toys to play with in their ResEdit adventures.

The end result of that was the Zen ROM, which became one of BMUG's most consistent sellers and encouraged people to buy the last copies of the third edition.

However, all good things must come to an end (or so we thought), and the third edition eventually went out of print. BMUG continued to sell its stock, but no longer could you go into a bookstore and ask for our book (though by that time you could ask for our other book, *The Tao of AppleScript*). Peachpit reverted the rights to us, and it became a joke with us that we would do a fourth edition.

Eventually, the joke started becoming a bit more serious, and we approached Hayden Books with the prospect of selling the fourth edition, which would be updated and bundled with an equally updated Zen ROM. They were interested, so we got to work.

Several months later, we are proud to present the fourth edition of *Zen and the Art of Resource Editing: The BMUG Guide to ResEdit*. Although updated in many respects, we hope that it has not lost its original purpose: to inspire Macintosh users to customize their computers beyond what the System offers by default.

Over the years, we have seen more and more new users come into the Macintosh community, devoted fans of their nifty computers. Many of these have become sophisticated power users, but the word *ResEdit* still makes all but the bravest quail. Visions of shattered hard drives, broken

System files, and general mayhem didn't seem to be significantly reduced by the temptation of glowing neon trash cans or funky patterns.

The purpose of this edition, as it has been in all other editions, is to educate people about how ResEdit can easily be used without fear of mass destruction. ResEdit is a tremendous utility for tweaking and enhancing your Macintosh, and when used with common sense, it won't do any damage.

It is a love of Macintosh that we, the authors, designers, and editors of this book, would like to impart to you. We all believe that useful information should not be withheld or held in the hands of a select few. Users should understand their Macintoshes and work with them more efficiently as a result. In essence, information is everyone's right.

As noble as that belief sounds, it is not original. In fact, it is the driving force behind BMUG. BMUG's slogan, in fact, is "We are in the business of giving away information." It is this motive that drives BMUG to continually put out a 400-page, semiannual newsletter. While most newsletters might contain information about company picnics or whatever, this newsletter informs you about technical and political topics, answers your most frequent questions, and always teaches you something you don't know. It is in that same spirit that these three BMUGers produced a book to inform the public about using ResEdit. We could not have produced this book without BMUG's willingness to promote it. Certainly the funding was important, but it was even more important to have BMUG's belief in the book—not just because it would make some money (very few people thought it would for the early editions), but because our book could only benefit from being presented by an organization that was built on a reputation of helping people. If you haven't looked into BMUG, we'd like to suggest that you do. For more information about BMUG, check out the pages right before the Index of this book or call our business office at (510) 549-2684. They'll be glad to send you some additional information about the group.

Derrick Schneider

Hans Hansen

What Is ResEdit, and Why Zen?

If you're a complete neophyte in the ResEdit world, you might be wondering why you should use ResEdit. You may have heard tales of its ability to bring down entire hard drives with a single mouse click, or something similar. Why take a risk like that? Well, to start with, those scary stories just aren't true.

The real reason to use ResEdit lies at the heart of the Macintosh itself. As Mac users, most of us appreciate the ability we have to customize the Mac to our needs. We can change keyboard or mouse speeds, change our background patterns, or even "view by" however we want to. If we don't like the way Apple does something, we can change it, either in the control panel or in the Finder menus.

ResEdit extends this ability to a grand scale. With ResEdit, if you don't like an icon, you can just change it. If you want a new sound, you can use ResEdit to install it. If you want a whole new keyboard layout, just change it with ResEdit. Don't just work on your Mac, make it work the way you it want to!

Incidentally, we've never heard of someone erasing their hard drive with ResEdit, so don't be too worried about it. If you pay attention to what you're doing and don't recklessly delete resources or anything like that, you'll be fine.

When we started this book, we had to figure out who our audience was. This is important, because ResEdit was designed as a tool for developers who write programs on the Macintosh, but this book was not to be aimed at developers. ResEdit gives one the power to quickly create the resources which are so common among Macintosh applications—which in fact are the roots of the Macintosh interface. However, we're not Macintosh programmers, and we use ResEdit all the time. We used it to change our trash cans, edit menus, change the watch cursor, and a variety of other tasks. We know of lots of people who, like ourselves, used ResEdit to "tweak" their programs in subtle ways. As a result, we decided to aim the book at regular people who use their Mac for both entertainment and work.

Programmers may still want to buy it as well, because it contains a lot of information about certain resources, but it will not tell you about the internal workings of ResEdit itself. For that, I highly recommend Apple's own *ResEdit Reference Manual*, available from Addison-Wesley. It contains a lot more of the information programmers use. In addition, those of you who are somewhere between developers and intermediate users of ResEdit might want to take a look at Peter Alley's *ResEdit Complete*, also from Addison-Wesley, to see if it is more to your liking.

Most chapters of this book cover these general areas: an introduction, instructions for using the resource editor in question, technical information (where applicable), and a final section with a step-by-step overview of the procedures covered, a summary of safety tips, and the Buddha's Guide to ResEdit Enlightenment, where you'll find tips and additional "exercises" you may wish to try. If you already know about what a resource does, skip ahead to the section dealing with the editor itself. Or, you may wish to go to the end of the chapter for new ideas about using a particular resource.

Some of you may be wondering about the title of this book. It is based on the title of the book *Zen and the Art of Motorcycle Maintenance*, which has become almost a shibboleth in the English language. The word *Zen* evokes images of mystical objects and people. A Zennist might be expected to talk about gateways into other planes of existence and powerful amulets. In short, Zen represents the mysterious and the bizarre. However, the philosophies of Zen, like many other philosophies and religions, also try to break very confusing concepts and thoughts into individual Truths. To a Zen practitioner, there are certain fundamental Truths that rule the Universe and all things in it.

Both of these definitions of Zen apply to the subject of this book. ResEdit has always been something of a mystical conundrum to the uninitiated. It may seem to occupy a different plane of existence, one that requires a deeper understanding of how a Macintosh works. As with Zen philosophy, one might hear a variety of ideas about the truths of ResEdit; some of these might be correct, but some are incorrect as well.

Zen and the Art of Resource Editing

This book, however, emphasizes the second definition of Zen. We have tried to break down ResEdit into simple truths that can be applied to many other situations in the computer world. We hope this book gives you the confidence to experiment with other resources as well.

Yes, ResEdit can destroy your System and many other things. However, if one is careful and follows certain rules, it is possible to become something of a ResEdit guru.

And, having attained such enlightenment, you'll want to share this experience with others. This is indeed the BMUG way.

About the Zen CD-ROM

Most computer books come with a floppy disk with some interesting software. Well, there's so much cool stuff out there that we couldn't possibly whittle it down to fit on one, or even two, floppy disks. So, we decided to do something different: This book comes with a CD-ROM.

On it, you'll find tons of snazzy resources, dozens of nifty utilities, and of course such essentials as ResEdit itself. We recommend you explore the CD-ROM in true BMUG style; lock yourself in your room, close all the blinds, and spend three sleepless weeks looking at everything! If you don't have the time (or lifestyle) to do this, you might want to take a more leisurely approach.

Open a couple folders, install a couple utilities. Take your time. There's an unbelievable amount of stuff to discover, and you will certainly not exhaust this collection anytime soon (don't be put off by the seemingly small size of the CD-ROM; icons don't take up much space!).

There's too much on the CD-ROM to give a complete list of what's on it. In the back of this book and on the disc itself, you'll find a description of some of the highlights, and that should get you started. You can also check the CD Highlights folder on the disc for aliases to some of our favorite resources and utilities. The rest you'll have to discover on your own!

Don't have a CD-ROM drive? Well, you'll miss out on the sheer mass of stuff on the CD-ROM, but there is a floppy available with some of our favorite stuff on it. Just call BMUG at (510) 549-2684, and ask for the almost-as-cool Zen Floppy Disk. If you're eager to get started right away, ResEdit is available from some other sources, most notably via anonymous FTP at [ftp.support.apple.com](ftp://ftp.support.apple.com/pub/apple_sw_updates/US/Macintosh/Utilities) in the `pub/apple_sw_updates/US/Macintosh/Utilities` directory. Don't have Internet access? Check out your local user group and see if they have a copy of ResEdit they can get you (you should check out your local user group anyway; they probably have a lot of interesting stuff you won't be able to live without).

Using the Zen Apple Guide

If you're running System 7.5 (or higher), you get an added bonus from the CD-ROM. On it you'll find the Zen Apple Guide file. To use it, copy it to your hard drive in the same folder as ResEdit (it's in the same folder as ResEdit on the CD-ROM). That's it. In order to get to it, you'll have to launch ResEdit first. When you do, it will be at the bottom of the Help menu.

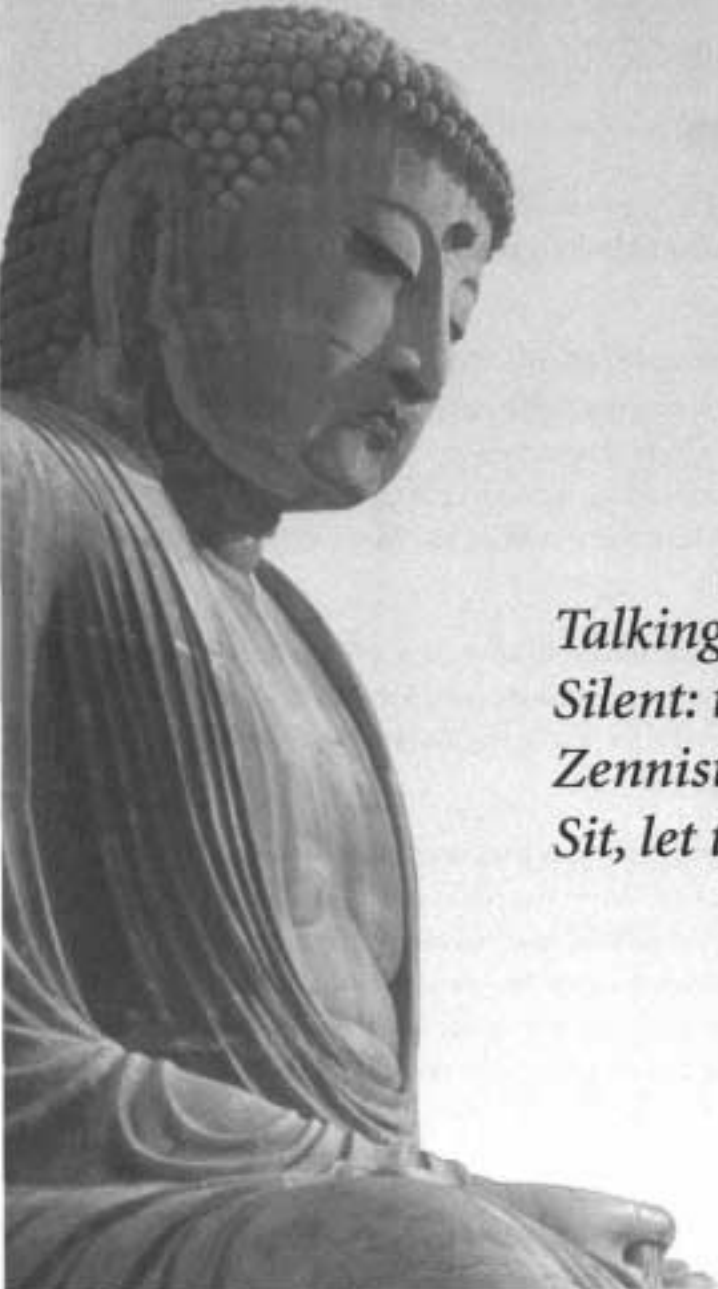
With the Apple Guide file, we've provided all the step-by-step stuff we've written in the overview sections at the end of each chapter. This way, you'll have a handy reference for some of the most common tasks you'll do with ResEdit.

How do you edit an icon? How do you add a Command-key to a menu? Don't feel like flipping through the pages of the book? No problem. Just open up the Apple Guide and you'll have instant access to the information, right there online. If you're a mobile user, you don't need to carry the book around everywhere you go: The pertinent information is all at your fingertips. Need we say more?

This is cool, and you should check it out.

Chapter Zero

Jumping Ahead



*Talking: seven steps, eight falls.
Silent: tripping once, twice.
Zennist everywhere,
Sit, let the mind be.*

Zen and the Art of Resource Editing

Feeling a bit antsy? Don't want to understand what you're doing before you do it? Think you're smarter than everyone else who's going to pick up this book? Maybe you just think that we wouldn't tell you anything that would screw up your computer, so why not just ignore the lengthy explanations and look for clues to elusive resource hacks?

Well, if you find yourself asking any of these questions, just shut off your brain and read the following instructions—they may or may not be interesting, and they may or may not be dangerous.

Installing ResEdit

Before you shut off all your reasoning powers and succumb to our control, there is one little thing you need to do—install ResEdit.

Unlike so many software applications these days, you won't need to spend eons fumbling through a stack of floppy disks or centuries determining which options to select when you “install” ResEdit. The application is small, it doesn't put any weird extensions in your System Folder, and it runs on any Mac with a minimum of RAM.



ResEdit 2.1.3

If you have any trouble
finding it look for this icon.

To install ResEdit, all you need to do is copy it from the CD-ROM that came with this book to your hard drive. Just the application and nothing else. It is located in the ResEdit folder on the top level of the CD-ROM.

You can put it anywhere on your hard drive. In fact you don't even have to put it on a hard drive; you could run it from a floppy disk or straight off the CD-ROM. However, this would be much slower, and why do you have a hard drive anyway if you're not going to use it? We like to put ResEdit on our desktop so that we can get to it easily; right down there next to the trash is very accessible.

If you have any trouble with this or don't have a CD-ROM drive, check the very last page of this book for some options.

Backing Up Your System

As much as we'd like to have you go through this chapter without a net to catch any loose mousing, good conscience forces us to show you how to do what we always ignore.

You should always have a complete backup of your computer should anything go terribly wrong. And no, we're not just talking about using mind-altering System hacks; this could happen to you quite innocently. Let's say you or your two-year-old son, daughter, or puppy does something less-than-sanitary to your keyboard. This may cause your Mac to think you just told it to erase your hard disk, and your Mac might even manage to think that you clicked all the appropriate warnings to make it really happen. Well, if it does, you'll be glad you have a backup.

For the purposes of the fun adventures in this chapter, though, we only want to make sure you back up your System file; the rest of your hard drive is fair game.

Step 1: Open the System Folder on your hard drive.

Step 2: Locate the System file (the one named simply System that looks like a suitcase).

Step 3: Hold down the Option key while dragging it to your desktop.

This will copy the file instead of merely moving it.

You're done. When you discover how much you hate what we're going to do to your Mac, you can use this copy of the System file and the steps at the end of this chapter to set everything back to normal.

Mystery Spot One

Now that you've got ResEdit nice and handy and an emergency backup of your System, release yourself into our control. We promise to subvert you and your Mac's System with the power of customization!

Step 0: Launch ResEdit (by double-clicking the icon).

Step 1: Click the splash screen real hard to make it go away faster.

Step 2: In the open dialog, go to the Zen CD and into the Book Files folder. Open the file named QuickStart RSRCs.

Step 3: When a window with lots of little icons opens real wide, press Command-A to select all the contents of the window, and then press Command-C to copy them.

Step 4: Now press Command-O to open another file.

Step 5: Go into the System Folder on your hard drive and open the System file (simply named System).

Step 6: Click OK—just do, ask not why.

Step 7: When another window with lots of little icons opens, press Command-V.

Step 8: Click Yes.

Step 9: Press Command-Q to quit ResEdit.

Step 10: Repeat Step 8 and then go on to Step 11.

Step 11: Choose Restart from the Special menu.

Can you figure out what we changed? Did you blindly install all our resources over the nice ones you had before? Good. The point here wasn't to learn about the resources, but to get a feel for interacting with ResEdit. It really isn't hard, is it? It works just like most friendly Mac applications.

But maybe you want more? That just wasn't exciting enough for you? After all, you've used the Copy and Paste commands a thousand times. So now you want us to really go where you've never gone before. Well, hang on to your mouse, you're going in!

Mystery Spot Two

Okay. This time we're going to make a little edit to your System file. It won't seem like much...at first.

Step -10: Launch ResEdit (just like before).

Step -9: Click on the clown's left ear.

Step -8: Use the open dialog to open the System file again.

Step -7.5: Click OK.

Step -7: Skip Step -6 and go straight to Step -5.

Step -6: Breathe.

Step -5: In the huge window with lots of icons, double-click the font suitcase icon—it's the one with the letters FOND underneath it.

Step -4: Press the Return key (only once).

Step -3: Scroll the window all the way to the end.

Step -2: Select the 12 in the field named Res ID.

Step -1: Type 396 alternating between your left and right hands.

Step 0: Press Command-Q.

Step 1: Click Yes.

Step 2: Restart your Mac.

So is there anything wrong? Did you figure this one out too? Boy, you are as smart as you thought you were. There's no putting

Zen and the Art of Resource Editing

anything over on you. I guess we'll just have to stop trying. Yeah, right. (If we were actually too subtle for you, just look at the menu bar.)

Mystery Spot Three

Ready or not, here we go.

One Step: Launch the clown.

Too Step: Click the clown.

Three Step: Open the clown (really), okay?

Fore Step: Open the SND.

Me Step: Copy the pig.

You Step: Open the System, okay?

Wee Step: Paste the pig (anywhere).

Big Step: Quit the clown, yes?

Up Step: Open the Sound control panel.

Down Step: Play the pig.

Got that? I knew you could.

Mystery Spot Four

So now that you are getting tired of the silliness, we'll try to be more serious—just don't ask us to keep it up for long. Remember, the secretary will disavow any knowledge of you or your ResEdit force—good luck.

Step 1: Launch ResEdit yet again.

Step 2: Open the System file and click OK.

Step 3: Open the PAT icon.

Step 4: Open number 17.

Step 5: Select all by pressing Command-A.

Step 6: Press Delete.

Step 7: Click down in the black rectangle at the bottom left of the window and drag the pointer to the very last pattern in the bottom row all the way to the right.

Step 8: Let go of the mouse button right here.

Step 9: Select the pouring paint bucket tool by clicking once in the right column, down three squares, on the grid at the left side of the window.

Step 10: Click once in the blank white square area in the upper middle of the window.

Step 11: Press Command-S.

Step 12: Press Command-W twice.

So, did you figure out what we did to your System that time? If you did, you can go on to Step 13. If you didn't, try scrolling around.

Step 13: Press Command-Q.

A Mystery No More

Just like all good mysteries, there is always an explanation, and a conclusion—welcome to the “who did it” section of the chapter. You did it. With ResEdit. In the [put the name of the room you're in here—the room we're in is simply called “ugly brown”] room.

Using ResEdit is easy, isn't it? The trick is to know what you're doing and why it works. That way, when we run out of cool things for you to try, you can find and invent some of your very own.

Zen and the Art of Resource Editing

This book will do that and a lot more. We know you won't look at your Mac the same way when you're done. We also think you'll have a lot more respect for the complex and original operating system a few people invented back in 1984. For more on that, let's go to the next chapter.

Oh wait, some of you are probably looking for this last little tidbit first....

Resurrecting Your System

Don't like the way we butchered your System? Think you can do better? Well, you probably can and will, so rather than have you build on the lovely designs we've inspired you with, let's put back the old unmodified System we set aside at the beginning of the chapter.

Step 1 of 4: Open your System Folder (in the Finder).

Step 2 of 4: Drag your System file to the Trash.

Step 3 of 4: Drag the copy of your System we put on your desktop into your System Folder.

Step 4 of 4: Restart.

If this didn't work, you didn't follow our instructions to the letter, so it isn't our fault. However, here are a couple other things you can do in an emergency.

If your Mac can't start up—if it displays a flashing question mark in the middle of your screen—then start up using the Disk Tools floppy disk that came with your Mac (unless you own a Performa, in which case you made a Disk Tools disk from the image preinstalled on the hard drive, right?).

This situation means that your Mac couldn't find a usable System in whatever it thinks is your System Folder. This could be because you took out the System file but didn't put in the backup copy we made. Or because there was a freak accident between a particle of dust and a gamma ray that tweaked a bit of magnetically stored information on the platter inside your hard drive.

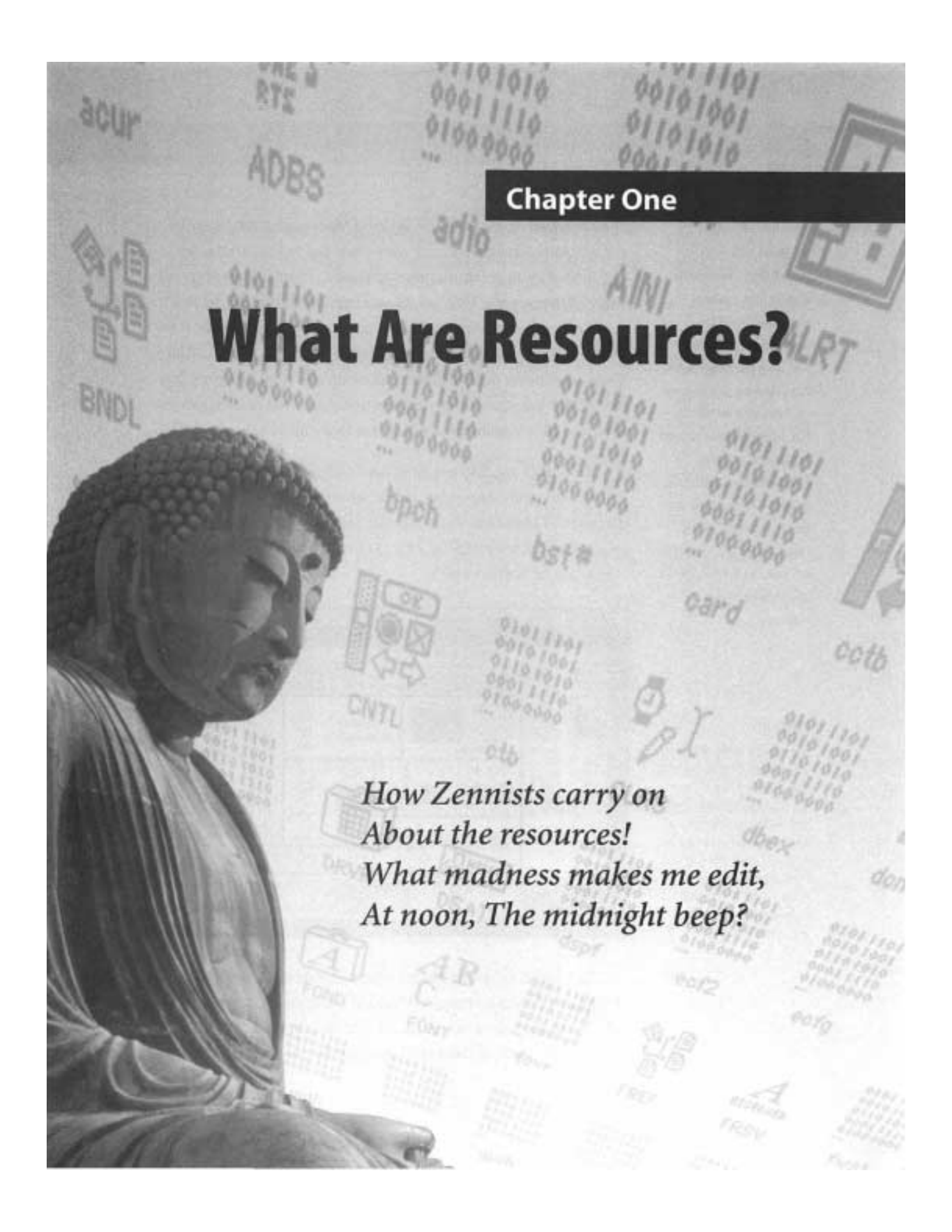
Once you've started up on the Disk Tools floppy, you can rearrange the System files, putting only one of them in the System Folder like you should have the first time. Or you can give up and reinstall your System from the original System disks.

If you really don't know what to do, consider calling BMUG at (510) 549-2684, becoming a BMUG member, and then calling the BMUG Helpline for emergency assistance.

But none of this is going to happen because you're going to follow our instructions carefully, and we are professionals. Well, one out of two isn't bad.

Let's go on!

Zen and the Art of Resource Editing



Chapter One

What Are Resources?

*How Zennists carry on
About the resources!*

*What madness makes me edit,
At noon, The midnight beep?*

Zen and the Art of Resource Editing

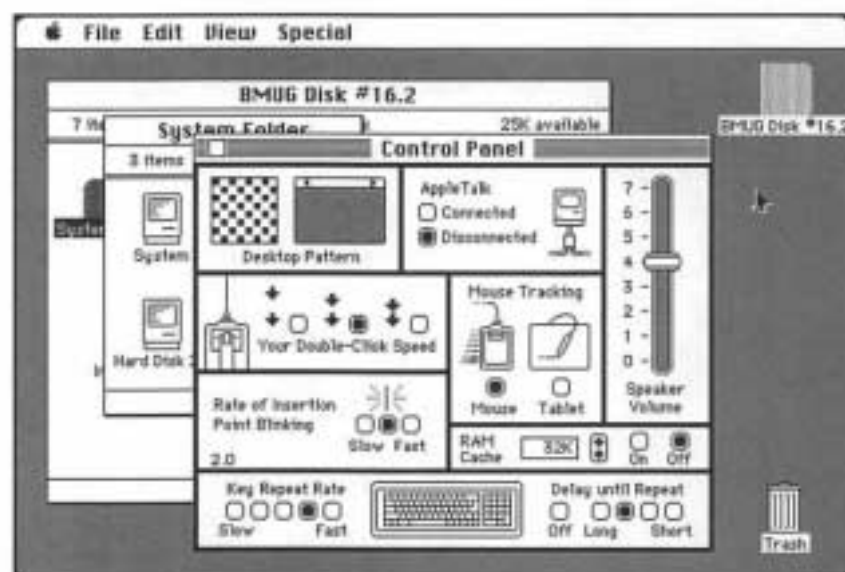
The Mac of 1984 really isn't much different from the Mac of today from a resource perspective. While the appearance and interface have improved with time and there are many new resources, the interaction among the Mac, its System resources, and applications' resources haven't changed much at all. What does this mean for the Mac of the future? Well, it could look and feel completely different and still be built on a similar foundation of resources.

Figure 1.1

The desktop of the original Mac isn't much different from that of today's Mac. From a resource point of view, the similarities are even deeper. While the contents of the resources have changed and there are many new resources, all the original resources are still a part of the latest System.

In 1984 when Apple first unveiled the Macintosh, computer users and software developers alike were attracted to the innovative elegance of its now familiar user interface. There is a great consistency in the way the Mac "looks and feels," regardless of what program you are using. Resources are part of the reason for this. They are a pervasive, hidden aspect of what makes a Mac a Mac. Most users never deal with them directly. It is up to the programmer of the software you're running to define what resources are used and how you interact with them (see Figure 1.1).

Resources are discreet aspects or components of the software you run on your Mac—they define what it looks like and how it is displayed. Think of resources as the ingredients in a fine recipe, most of which are essential, others of which are optional, and all of which can be fudged a little.



To make its user interface easier to create and modify, the designers of the Macintosh defined some standard resources with which programmers can quickly build their applications (see Figure 1.2). Each of these resources can be individually modified

with ResEdit. This is an important part of why the Macintosh “looks and feels” the same in any program.

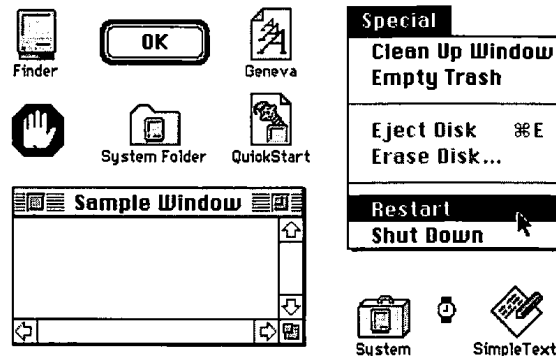


Figure 1.2
Elements of the Mac interface. All these items are made from resources.

Every element of the Mac interface that we have come to know and love is a resource. These elements include menus, icons, cursors, keyboard layouts, and many more. Any file or program running on a Mac has the potential to contain resources, in a section of the file known as the “resource fork” (see Figure 1.3). The other part of a Mac file is called the “data fork” and is where the actual numerical and textual data (such as the text and usually the formatting information in a word processing document) lives.

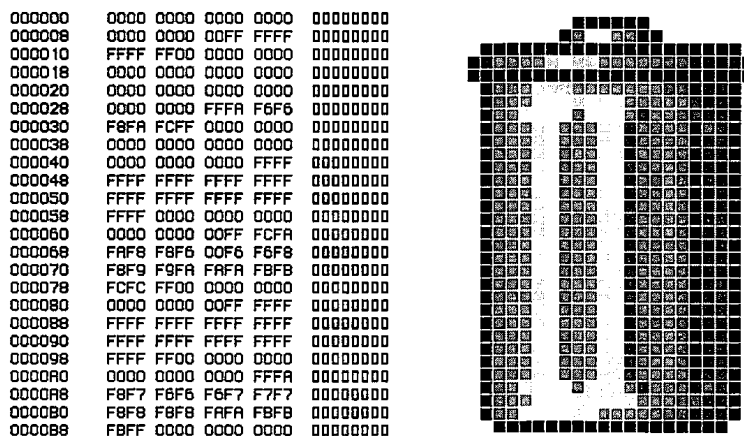


Figure 1.3
A representation of the actual data of the trash can icon's resource.

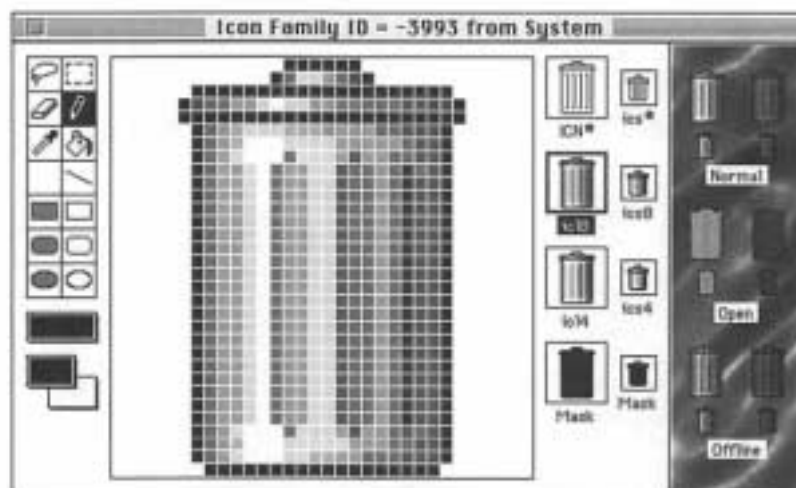
Within the resource fork, each resource has several identifying markers. These include the type of resource, the ID number of the resource, and, sometimes, the name of the resource.

Zen and the Art of Resource Editing

Of course, because we are dealing with a computer, this data is really just a stream of incomprehensible numbers. In the early days of Macintosh, creating or changing a resource usually involved a calculator and a large amount of caffeine. Thankfully, some engineers at Apple decided to use their calculators and caffeine-saturated carbonated beverages to make creating and editing resources less of a chore. When they were done, a new age dawned, birds sang, flowers bloomed, and ResEdit 1.0 was born. ResEdit is short for "Resource Editor," and it enables you to, as its name implies, edit some of these common resources graphically (see Figure 1.4). The original ResEdit was very dangerous and did little to help the average user. New versions continued to come out, each with significant improvements. Later, as System 7 became more and more of a reality, Apple realized that it was going to need a new ResEdit that could handle all the new resources in System 7. The most current result of that is ResEdit 2.1.3.

Figure 1.4

ResEdit enables you to edit icons in a nice painting environment instead of editing the hexadecimal code displayed in Figure 1.3. You guess which is easier.



How Do They Work?

When a program needs a resource (for instance, when the Finder puts a menu in the menu bar), the resource is not created on the spot. This would take a lot of time and seriously affect the

performance of a Macintosh, because the Mac is very resource-oriented. We all know that the Mac moves pretty quickly when it wants to, which it can do because it uses stored resources to enhance speed.

What really happens, then, is that the program realizes that it needs to put menu #5 into the menu bar, so it looks at its resource fork and pulls out menu #5. It doesn't look at the menu to see if it says what it should; it just grabs all the info and throws it into the menu bar (see Figure 1.5).

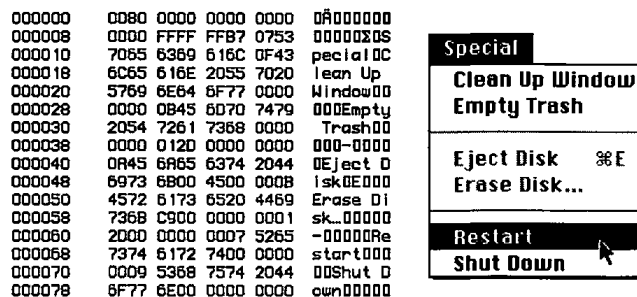


Figure 1.5

On the left is the code that makes up the Special menu on the right.

To create this resource, the authors of the program can use ResEdit. They can then redesign the resource as needed without really changing the program, because the Mac will just display whatever's in menu #5 when the program runs.

Just as the programmer uses ResEdit to create and fine-tune resources, you can use ResEdit to modify them yourself. With ResEdit, you can design your own menu #5 and replace the one in the program. Again, the Mac won't pay attention to what's in the resource, it will simply use your redesigned menu instead.

Despite all the weird codes between the names of the menu choices in the middle, this resource contains only the menu title, menu items, and command keys. It does not have any information for the computer to use to figure out how to draw the lines, type, or interaction with the cursor.

Looking at a Program's Resources

Before you get your hands dirty with actual resource editing, it's important to spend some time getting familiar with ResEdit.

If you haven't copied ResEdit to your hard drive already, insert the CD-ROM that comes with this book and open the ResEdit folder. Copy just the ResEdit 2.1.3 application, putting it anywhere on your hard drive. Because it is only 700K, you don't have to put it on a hard disk if you don't want to. However, running it from a floppy disk or off the CD-ROM would be much slower. If you don't have a CD-ROM drive and need more information on getting ResEdit, see the last page of this book.

Launch ResEdit (by double-clicking its icon, of course). The first thing you see is ResEdit's splash screen, which you can quickly get rid of by clicking anywhere in it (see Figure 1.6).

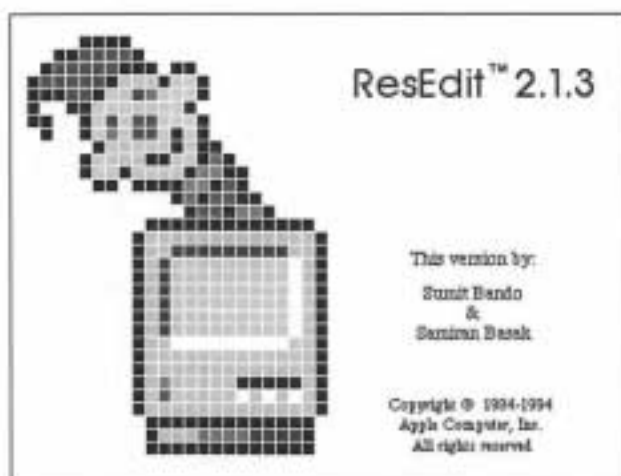


Figure 1.6
ResEdit's splash screen. You can click anywhere on it to make it disappear.

After that, ResEdit gives you a standard open file dialog.

Because the Mac is so heavily resource-based, let's start by looking at the heart of it: the System file. In general, the System file contains more resources than any program you'll find. However, it's also the most sensitive: A change to your System file will affect everything else on your Mac. At the moment, though, we're just looking, not touching, so you don't need to worry about this. If you followed the text in Chapter 0, "Jumping Ahead," you've already seen what modifying the System can do.

Use the open file dialog to select the System file in the System Folder (see Figure 1.7).

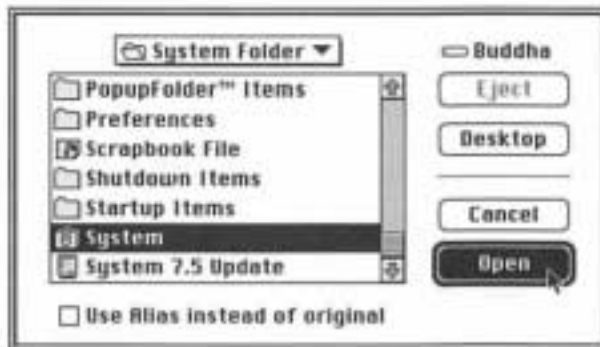


Figure 1.7

ResEdit's open file dialog box with the active System file selected.

Because the System is so sensitive, ResEdit quickly puts up a warning about working on the active System. Again, we're just looking and not making any changes, so you can click OK (see Figure 1.8).

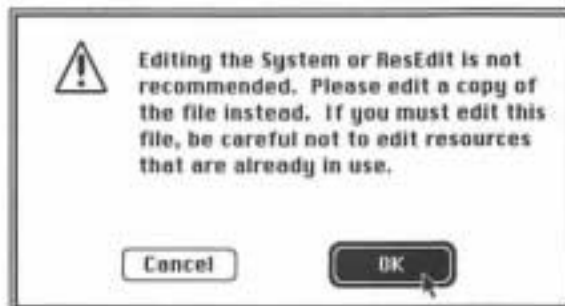


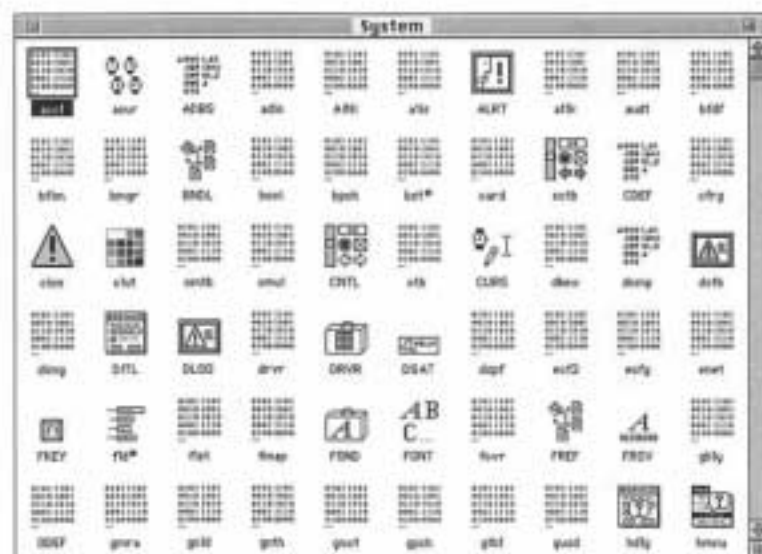
Figure 1.8

ResEdit's warning that you shouldn't edit the active System. We normally agree, but in this chapter we're just looking and not changing anything, so it is okay to continue.

ResEdit then displays a window (named System) containing icons that represent the types of resources that are in the System file. By scrolling down, you'll see that there are a lot of resources to look at (see Figure 1.9). Many of the resource types have distinct icons that are somewhat descriptive of the resource types. For instance, the icon for the CURS resource type implies that this resource type is used to describe cursors. Seeing this icon in the System file means that the System file contains one or more cursor resources.

Zen and the Art of Resource Editing

Figure 1.9
The System file opened in ResEdit. The accel resources are selected.



Many of the resources, however, do not have distinct icons, but instead have icons that look like groups of 0s and 1s. These are resource types that ResEdit doesn't recognize. ResEdit can still open these resources, but it doesn't always provide a nice graphical editor like it does for many of the resource types that have distinct icons.

The Buddha's Revelations

You can always use ResEdit's preference commands to change what ResEdit does when it launches.

You can double-click any of these icons to see the resources of that type. For instance, scroll down until you see the ICON icon, and double-click it (see Figure 1.10).

A window like this, which shows all the resources of a particular type, is called a picker. In this particular picker, you can see all the ICONs that are in the System file (as the name of the picker window states). Each one has an ID number underneath it. Double-click the ICON with the ID number 2 (the triangle with the exclamation point). This opens the ICON with the ICON editor (see Figure 1.11).

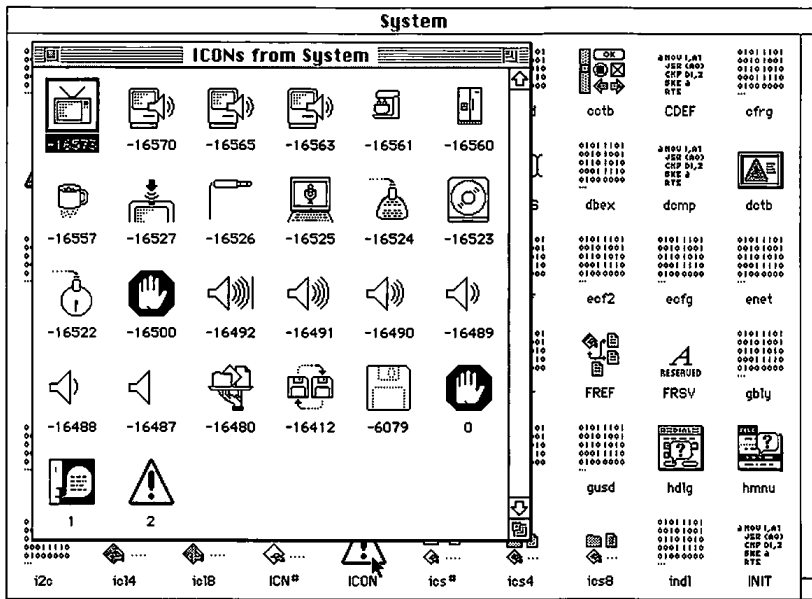


Figure 1.10
The ICON picker window displaying all the ICON resources in the System.

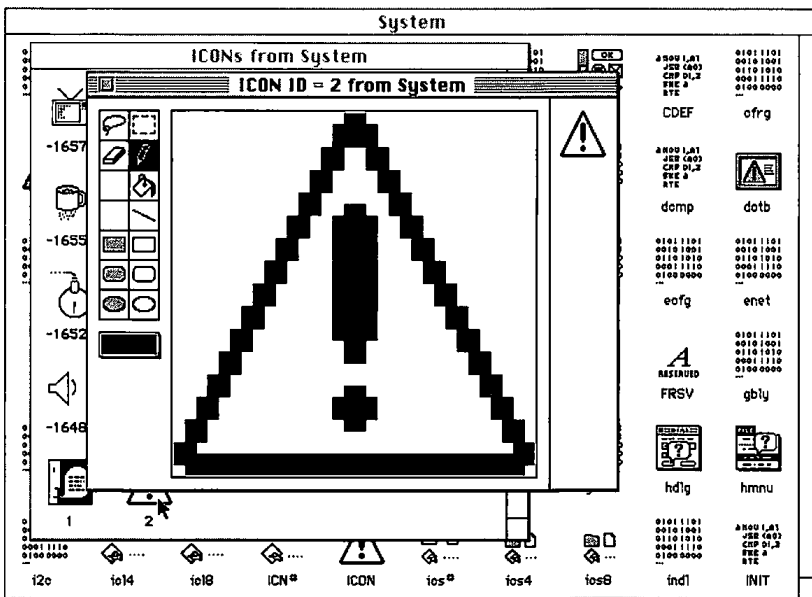


Figure 1.11
The ICON editor window displaying ICON resource ID 2.

Remember that we're not making any changes yet. If you were to make changes in this ICON resource, alert dialog boxes would be changed universally because this icon is commonly called by programs that want to get your attention. For the moment, close

Zen and the Art of Resource Editing

the ICON editor window and the ICON picker window, so that you are again looking at the resource types in the System file.

Some resource types, like sounds, aren't visual, but ResEdit still recognizes them. Open the 'snd' resource picker by double-clicking the 'snd' icon (see Figure 1.12).

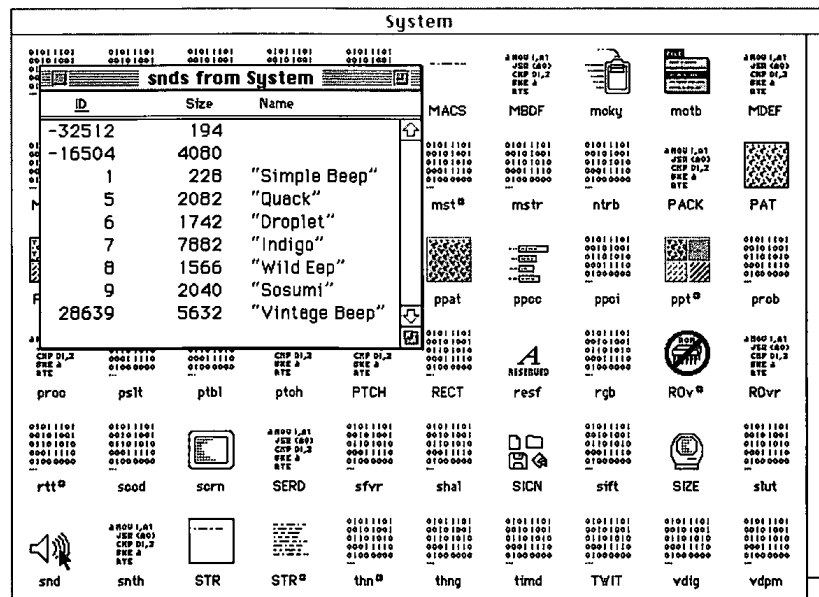


Figure 1.12

The 'snd' resource picker window from the System file. Your own copy may be different depending on the sounds you have.

This picker shows a list of the 'snd' resources in the System file. Each item in the list is a different 'snd' resource. For each 'snd' resource, you can see the ID of the resource, the size (in bytes) of the resource, and the name, if it has one. Every resource of a particular type has a unique resource ID. A resource can have an optional name—usually just to provide an easier method of recognizing it (for humans); programs rarely refer to resources by their names.

You can hear what each sound resource sounds like by selecting it and choosing Try Sound from snd menu. For instance, 'snd' number -16504 is the "camera click" that is used when you take a screen shot by pressing Command-Shift-3 (see Figure 1.13). Most

of the others are the beeps you can choose in the Sound control panel.

ID	Size	Name
-32512	194	
-16504	4080	
1	228	"Simple Beep"
5	2082	"Quack"
6	1742	"Droplet"
7	7882	"Indigo"
8	1566	"Wild Eep"
9	2040	"Sosumi"
28639	5632	"Vintage Beep"

Figure 1.13

Selecting a 'snd' resource to try using the Try Sound command in the snd menu.

Now that you're familiar with how to use ResEdit to open programs and look at different kinds of resources, you'll need to close the System file without saving any changes you might have made. Close the 'snd' picker window and then close the window containing all the System's resources. If you have inadvertently made some changes, ResEdit will ask you if you want to save your changes. The correct and only answer to this is No (see Figure 1.14). If you haven't made any changes—come on, you were tempted to see what would happen, weren't you—you won't see this dialog box.



Figure 1.14

Don't save any changes you may have made. We'll get to editing soon enough.

Resources Are the Foundation of the Mac

You've seen that a resource is a chunk of data, just like a word or a picture, and that the Mac uses these chunks of data to determine what to display in the interface of any and all programs,

including the Finder. Once you understand these chunks of data, you can change them so that they behave the way you want. Many resources have an intuitive name and editor, so that you can easily find the relevant resources and edit them quickly. By exploring these resources and their editors, you will be able to change the way your desktop looks or change the actual functionality of your programs.

Resources Overview

Discovering resources in your Mac's System and understanding how they relate to everything you work with on your screen has been your first step. Here are some of the things that you'll want to remember so that you can explore on your own.

The Buddha's Revelations

Holding down Command and Option as you choose About ResEdit from the Apple menu gives you a secret dialog containing the names of the ResEdit team.

To see what resources are in a file:

- Launch ResEdit.
- Dismiss splash screen.
- Use open file dialog to select the file you want to look at.

To open a resource picker:

- Double-click the icon of the resource type.

To open a resource in its editor:

- Double-click the specific resource you want to edit in the picker.

To hear what a 'snd' resource sounds like:

- Select the 'snd' resource you wish to hear.
- Choose Try Sound from the snd menu (only visible in the 'snd' picker).

Safety Tips—What Not to Do!

Don't get scared. Most of these tips are going to sound a bit mothering. You know, the "don't save your changes until you've washed behind your ears" kinds of things. There really aren't any commands that can screw up your System or cause you to lose data that don't warn you first. Just keep these tips in mind and you'll feel more in control of your own destiny.

- Don't work without a net! Make a complete backup of all your data and System and understand how to restore it in case of an emergency. Trust me. It will happen to you someday.
- Don't work when you're so tired that you can't read the warnings in the alert dialog boxes and think clearly enough to click Cancel.
- Don't select all the resources and press Delete.

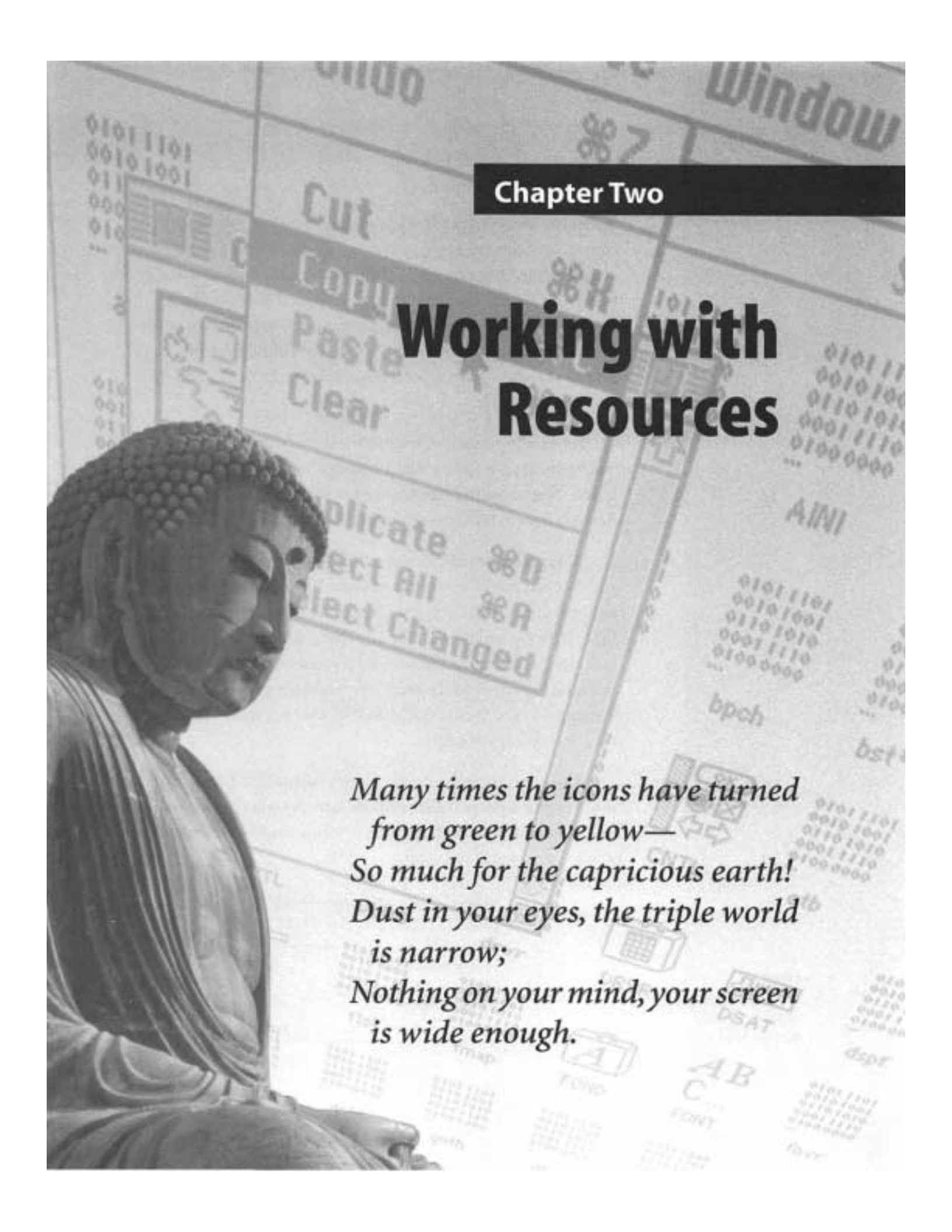
The Buddha's Guide to ResEdit Enlightenment

Throughout this book you will find this section of hints toward ResEdit enlightenment at the end of every chapter. You'll find interesting tasks through which to learn, hints at cool hidden resources, and sometimes a bit of silliness to lighten your day. Exploring the Mac is what this is all about—enjoy your adventures and watch out for Red Herrings!

- Can you launch ResEdit with one click? (And maybe a drag?)
- How do you set ResEdit to not show its splash screen?
- Can you find the resource in your System for controlling mouse speed?
- What do you think the DSAT resource does?
- Can you guess why ResEdit's clown has such big ears?

Zen and the Art of Resource Editing

- Can you figure out what every icon in the System is used for?
- Should you open the System and delete all its resources?
- What happens when you press Command-A and then Command-O?



Chapter Two

Working with Resources

*Many times the icons have turned
from green to yellow—
So much for the capricious earth!
Dust in your eyes, the triple world
is narrow;
Nothing on your mind, your screen
is wide enough.*

So far, you've used ResEdit only to view resources—unless, of course, you followed along with us in Chapter 0 where you actually edited resources. And that's what makes ResEdit useful to you; it can also be used to cut, copy, paste, and even duplicate these resources. As a result, you can move resources from one file to another, duplicate an existing resource within a given file, modify the way a given resources looks or works, or even delete a resource from a file.

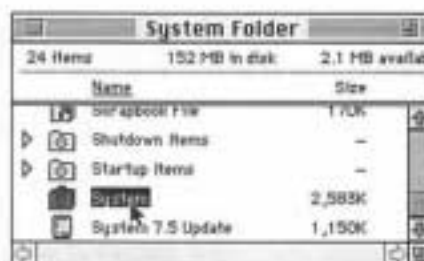
For example, you may want to make the “camera click” sound you found in your System file into a beep sound you can select in your Sound control panel. You can use ResEdit to do this, but first you have to take some simple precautions.

When you opened the System file previously, ResEdit warned you that you should not edit the System file, because it was currently in use. At that time, you were just looking at the resources in the System file and not making any changes. But now, you will be making changes to the System file, so you should follow the advice in the dialog box. ResEdit won't give you warnings about every file you open, so you should realize that it's always a good idea to work on a backup copy of a file.

If you are not already in the Finder, you need to first switch to it. Go to your System Folder and select the System file (see Figure 2.1). Choose Duplicate from the File menu, and the Finder creates a new file named System copy.

Figure 2.1

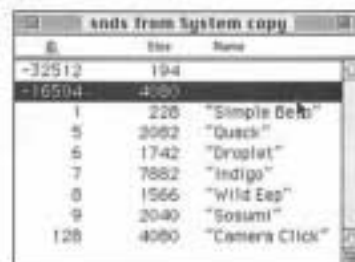
The System file selected in the Finder to make a duplicate before editing.



If you leave both System files in your System Folder, your Macintosh will be confused about which one it should use. To avoid this, move System copy to some location outside the System Folder (such as the desktop).

Open ResEdit, and use the Open command to select and open System copy. In the Finder, you can also drag the System copy icon onto the ResEdit icon, which causes ResEdit to open that file. Notice that this time it does not give you a warning dialog; this is because this file is not currently being used.

Scroll down until you see the 'snd' icon. Double-click it to open the 'snd' picker. You'll see the same list of 'snd' resources you saw in the real System, including the beep sounds and two unnamed resources at the top (see Figure 2.2). Select the resource with the ID -16504. You can verify that this is the "camera click" sound by choosing Try Sound from the snd menu.



ID	Size	Name
-32512	194	
-16504	4090	
1	228	"Simple Beep"
5	3082	"Quick"
6	1742	"Droplet"
7	7882	"Indigo"
8	1566	"Wild Eep"
9	2040	"Sosumi"
128	4090	"Camera Click"

Figure 2.2

The 'snd' resources in the System file.

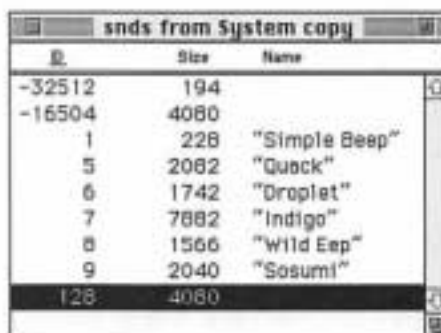
For a sound to show up in the Sound control panel as a beep sound, it must have an ID number that is 0 or greater and a name. You could change the resource's ID and give it a name, but then it would not be used by the System when making a screen shot, because the System needs a resource with this ID number. You

need to duplicate this resource so that you can set it up to be a beep sound, and leave a copy for the System to use when making a screen shot. With -16504 selected, choose Duplicate from the Edit menu. ResEdit will make a new 'snd' resource, giving it the default ID of 128 (see Figure 2.3).

Because this ID is greater than 0, all that's left is to give this resource a name. With the new resource selected, choose Get Resource Info from the Resource menu. This will open the Get Resource Info window (see Figure 2.4).

Figure 2.3

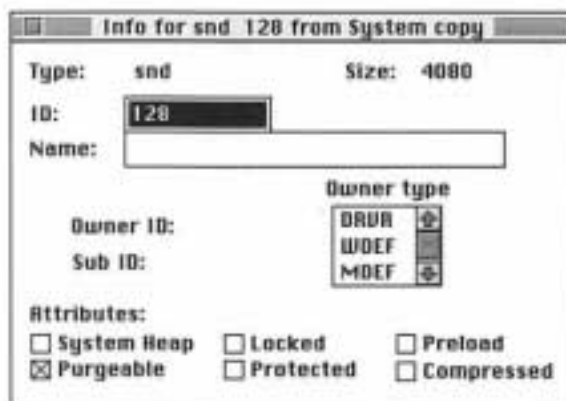
The second resource from the top duplicated as a new resource at the bottom.



ID	Size	Name
-32512	194	
-16504	4080	
1	228	"Simple Beep"
5	2082	"Quack"
6	1742	"Droplet"
7	7882	"Indigo"
8	1566	"Wild Eep"
9	2040	"Sosumi"
128	4080	

Figure 2.4

The resource information window for 'snd' resource 128 from the System.



Type: snd Size: 4080

ID: 128

Name:

Owner ID: Sub ID:

Owner type: DRVR, WDEF, MDEF

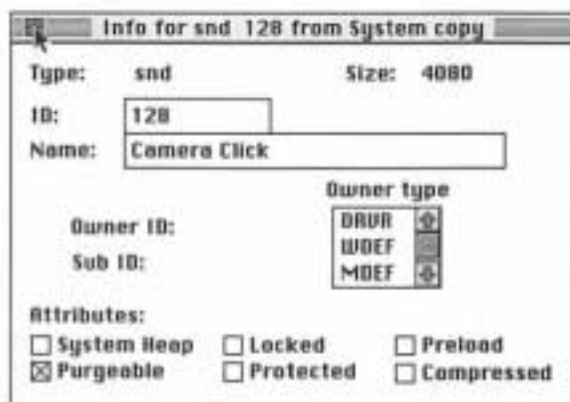
Attributes:

☐ System Heap ☐ Locked ☐ Preload

☒ Purgeable ☐ Protected ☐ Compressed

This window gives you a lot of information about a resource, but for the most part you'll be concerned only with the top two fields. These enable you to change the resource's ID and name. To give your new 'snd' resource a name that you'll be able to see in the Sound control panel, type the name you want to use in the Name field such as Camera Click (see Figure 2.5). When you've done this, close the Get Resource Info window. You'll notice that the resource is now shown in the picker with its new name.

Close all the open windows. When you close the System copy window, you will be asked if you want to save changes. And do you? But of course (see Figure 2.6).



More than one individual resource or resource type may be selected together to copy or duplicate. To select two contiguous resources, hold down the Shift key when you select them. To select noncontiguous resources, use the Command-key. If you use the Shift key to select two noncontiguous resources, ResEdit selects all the resources between those two resources. This can either be very time-saving or very frustrating if you do it unintentionally. You then copy and paste them, delete them, open them, or get info on them all at once.

Figure 2.5

The modified information for resource 128.

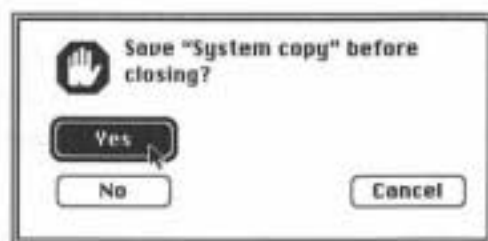


Figure 2.6

Saving your changes.

Remember that you've made these changes to System copy, not the actual System file. You need to replace the System in your System Folder with the System copy file.

To do this, first move the System file out of the System Folder (for instance, onto the desktop). Then, move System copy into the System Folder. Change the name of System copy to System. Make sure that you have a file named System in your System Folder, or your Mac will not start up properly. Restart your Macintosh.

Go to the Sound control panel, and you will see your new sound in the list (see Figure 2.7).

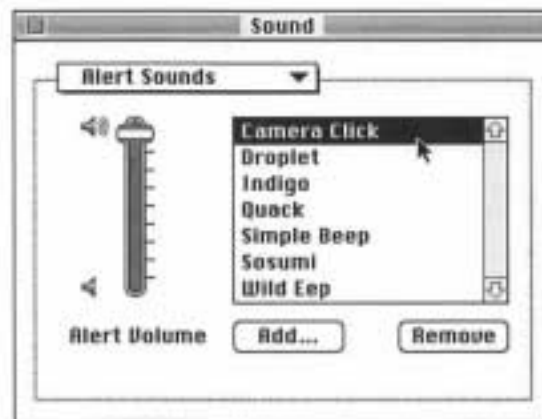


Figure 2.7

The Sound control panel with the Camera Click sound added.

Using Resources in Files

You've seen that the System file contains resources that are used by many applications on your Mac for simple, universal things such as alert dialog box icons and beep sounds. But applications can also use resources from many other places. Usually, most of the resources an application needs to function are stored in the resource fork of the application itself. But an application can also store its resources in files external to the program.

These external files can be organized just for the purpose of holding flexible resources that a programmer may want to change (or allow to be changed) easily without having to modify the application. A good example of this is a game that keeps all its 'snd' resources in a Sounds file. Alternately, resources can be

stored in documents to enable them to travel with the document to other machines, such as a HyperCard stack, which might contain custom ICON resources to use in a button or special 'snd' resources to play as appropriate. By storing those special resources in the document, the resources will be visible even on someone else's Macintosh.

Common resources you find in an application are resources that describe the icons you see on the desktop in the Finder, the menus you use when using the application, the actual code the application uses, and many other resources that define the unique aspects of the application.

Common resources you find in documents (or files) are resources that are unique to the data of that file and are stored in the resources fork of the file. An example would be a Photoshop document that uses a custom icon in the Finder that looks like the picture that is stored in the data fork of that file.

This resource management is very important to keep in mind if you want to distribute your custom resources. For instance, imagine that you're designing a HyperCard stack and want to use a custom icon. If you put that icon in your stack, that stack can use it for a button (it shows up when you choose to assign an icon to a button). However, if you want to use that icon in a lot of your stacks, it makes more sense to place that ICON resource into the HyperCard application itself. If you do this, however, your stacks won't show your cool icon when you give them to someone else, because their HyperCard application won't have the special resource you created.

Moving Resources

Getting your custom resource into an application or document with ResEdit is very easy. One of the simplest ways to move

the resource from one file into another is to copy (or cut) it from one document and paste it into another. To do this, you select the resource in its source document, copy it using the Copy command from the Edit menu (or by pressing Command-C), open up the destination document, and choose the Paste command from the Edit menu (or press Command-V). Once the resource is in the destination document, you can save your changes to make it permanent. You can also do things like change this resource's name or ID, which you already saw how to do in the previous section.

As a specific example of moving resources, you can use ResEdit to pull 'snd' resources out of places such as the System file or an application and place them into System 7 sound files that the Finder can play when you double-click them. We'll try this soon, but first there are a few things you'll need to know. For the System to recognize these 'snd' resources in these sound files, three things need to be true. First, the sound file and the 'snd' resource need to have the same name. You've seen how to change the name of resources already, so that won't be a problem. The other two important attributes of a sound file are that it must have a file type of `sfil` and it must have a creator type of `movr`. ResEdit will easily let you make those changes, too.

Types and Creators

On a Macintosh, every file (whether it's an application or document) has a particular four-letter code for its *file type* and another four-letter code for its *creator type*. The file type describes what kind of file it is, and the creator type names which application created it.

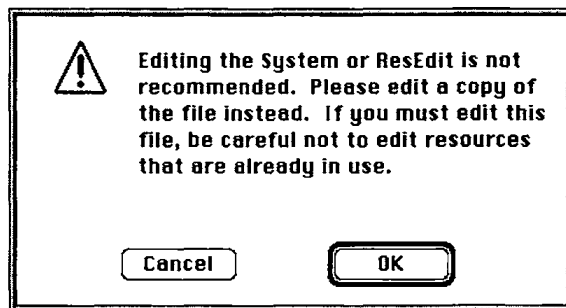
For instance, an application file has a file type of `APPL` and a unique creator type; in the case of SimpleText, its creator type is

ttxt (from the days when it was named TeachText). Every document that SimpleText creates will also have a creator type of ttxt, but might have any file type that SimpleText can create: TEXT for plain text files, PICT for pictures, or ttro for “teach text read-only” text documents.

Giant Sucking Sounds

With this knowledge, you’re ready to make sound files out of ‘snd’ resources you find in various programs. We’re going to take a sound resource from the ResEdit application itself and copy it into a System 7 sound file that you can double-click to hear.

Launch ResEdit, and use it to open the ResEdit application itself. You get a dialog saying that the System or ResEdit should not be edited (see Figure 2.8). Just like when you looked at the System in the previous chapter, you won’t actually be editing ResEdit, so it’s not a problem. You can get rid of the dialog by clicking OK.



The Buddha’s Revelations

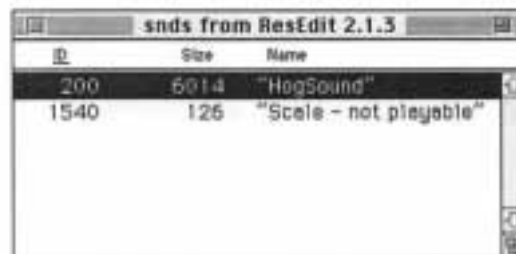
You might notice that the invisible icon files that store a folder’s custom icon have a blank type and creator.

Figure 2.8

If you’re just looking, click OK. If you’re editing, click Cancel.

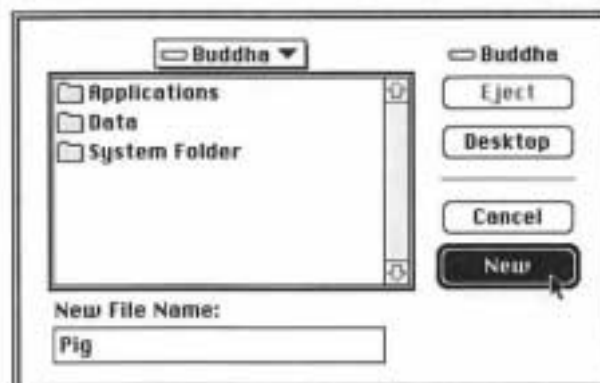
Open the ‘snd’ resource picker in the ResEdit window, and choose the ‘snd’ named HogSound (see Figure 2.9). Remember that if you want to hear what it sounds like, you can go to the Try Sound menu option in the snd menu.

Figure 2.9
The 'snd' resources
in ResEdit.



Once it's selected, choose Copy from the Edit menu. This puts the resource on the Clipboard, just as it would in any other Macintosh application. Go to the File menu and choose New. ResEdit asks you for the file name, which you can make anything you want (we chose Pig) (see Figure 2.10).

Figure 2.10
Saving a new resource file.



ResEdit then opens up the new Pig file's list of resources window. Because it's a new file, there aren't any resources in it, so the window is empty. Make sure this window is active and choose Paste from the Edit menu. ResEdit puts the resource into the file—you're able to tell because that window now has a 'snd' icon in it.

Open the 'snd' picker in that window and you see the single sound you just copied in (see Figure 2.11). Remember that for a System 7 sound file to work properly, the file must have the same name as the 'snd' resource. Obviously, the name of the file is not the same as the resource you just pasted in, but you can easily

change that. Choose the 'snd' resource and select Get Resource Info from the Resource menu. Change the name of the resource to Pig and close the window (see Figure 2.12).

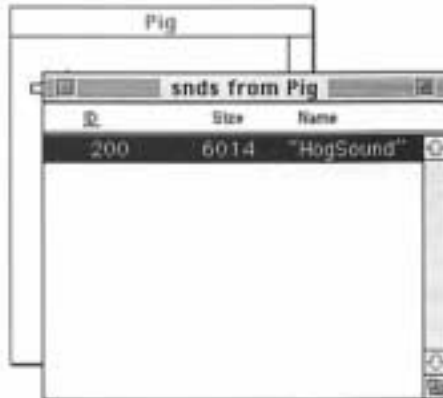


Figure 2.11

The HogSound from ResEdit pasted into the Pig resource file.

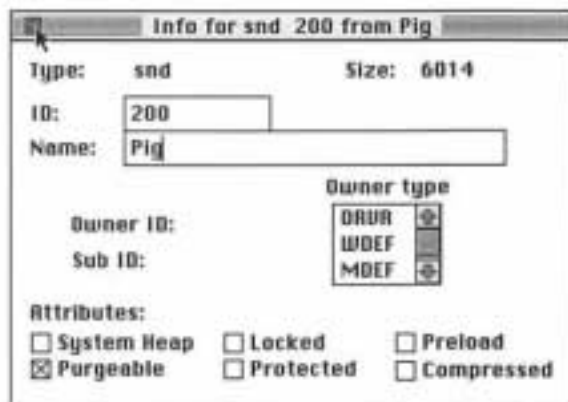


Figure 2.12

The name of the HogSound resource changed to Pig.

If you were to save this file now, it would be saved with a creator type of RSED and a file type of rsrc because it is a generic ResEdit resource document. You need to change this so that it's a System 7 sound file that the Finder will recognize. To do this, choose Get info for Pig from the File menu. This brings up a dialog that contains all sorts of information about the file (see Figure 2.13).

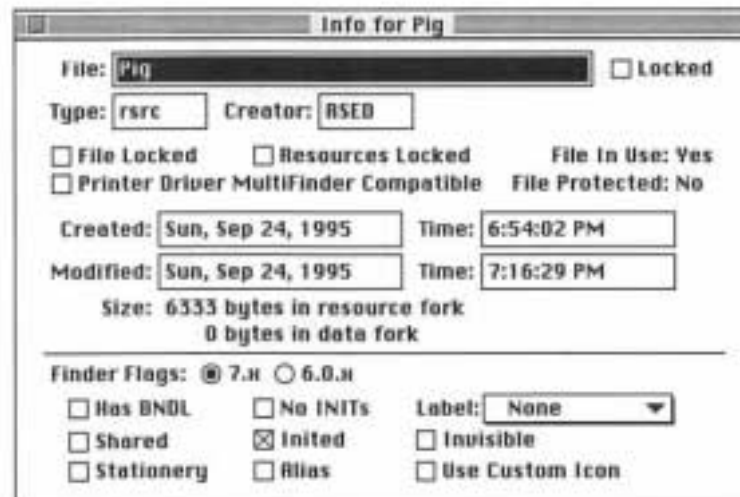


Figure 2.13

The file information window for Pig.

For more information of various file types and creators, check out Appendix B.



Figure 2.14

Double-clicking a sound file in the Finder plays it.

Though there's a lot of information here in this window, you care only about the fields labeled Type and Creator. Change the type to `sfml` and the creator to `movr`, and close the window. ResEdit then asks if you want to save the changes you made, to which you should say OK.

You are now done making the Pig sound file, so save the file by choosing Save from the File menu or by pressing Command-S, and close the window or quit ResEdit—make sure you save your changes when asked if you want to. Switch back to your desktop (in the Finder) and double-click the Pig file (see Figure 2.14). You should hear the sound of a pig oinking.

Other Resources to Move Around

Copying and pasting is a basic Macintosh operation, but when coupled with ResEdit's ability to change resource IDs and names, a lot of your resource "editing" simply entails moving resources from one place to the other. Whether moving resources from one file to another or copying the information in a resource to some other program, editing it there, and pasting it back over the original resource, some of the most interesting resource editing

isn't really editing the resource itself at all, but simply moving resources around.

PICT Resources

One resource that people often want to edit is a PICT resource. PICT resources are the usual way that programs store any pictures or graphics that they might use. For instance, if you look at the System's PICT resources, you see that one of them is the graphic displayed by the About Balloon Help menu item in the Help menu (see Figure 2.15). (Want to see for yourself? Open the System file with ResEdit. Open the PICT picker. Scroll until you see the right PICT.)



Figure 2.15

The graphic displayed in the About Balloon Help dialog.

However, ResEdit doesn't have a built-in PICT editor. Double-clicking a PICT resource in the PICT picker will open the PICT in its own window, but you can't do anything with it. Nothing, that is, unless you use the most powerful standard Mac commands, Copy and Paste. By copying the PICT from this window, you can paste it into your favorite paint program, modify it, and then paste the revised version (or some completely new version) back into the window.

One example of how you could use this technique to customize your Mac is to change the MacOS logo that is displayed when you start up your Macintosh with System 7.5.1 installed (yes, you can now easily date the production of this book). If you don't have

Yes, normally we would suggest you work on the copy as opposed to the original. But in this case, we know from our close friends at Apple that this file is used only during the startup process, and isn't actually in use when it's just sitting in your System Folder. (You can see this for yourself because when you open it in ResEdit, you don't get the warning about editing a file that is in use. That means this file isn't in use.) So, editing this copy isn't really any different from editing the backup copy, except that it is already in the right place for the System to find it when you restart.

System 7.5.1 installed because it is too new or too old for you, don't worry about it—it's the concept that's important. You can still follow along and apply this method to some other PICT you might want to modify.

You might expect the PICT you're looking for to be in the System file, but it's not. It's in the System 7.5 Update file in your System Folder. Make a duplicate copy of this file and move it outside your System Folder for backup, so if you mess up (which you won't), you can replace it.

When you open the update file (the original one in the System Folder) with ResEdit, and open the PICT resource picker, you can see that there are actually six different pictures of the MacOS logo. Which of these your System uses is based on the kind of machine you have. PICTs -16506, -16505, and -16504 are used for color, grayscale, and black-and-white large screens, respectively. The next three are used in the same way for smaller screens. You probably want to edit all the logos for your size screen, in case you ever start your machine in a different bit depth. We'll show you how to replace one of the PICTs, and you can use that information to replace the others. Because we wrote this book using a color machine with a large screen, we used PICT -16506. Double-click that PICT in the resource picker so that the PICT is opened in its own window (see Figure 2.16).

At this point, you could copy the PICT, paste it into your favorite paint program, and edit it to generate your own picture. If you're not artistically inclined, however, we've provided a file for you with a new and improved logo. It's called Zen OS.pict and you'll find it on the CD-ROM in the Book Examples folder.

Open the new logo file in your favorite picture editing program or double-click it and it will open in SimpleText (which is included on the CD-ROM for those of you who don't have 30

copies spread throughout your hard drive already). Copying the picture from a paint program (or viewer) is all you need to do for the purpose of this exercise.



Figure 2.16

The MacOS PICT resource used when starting up under System 7.5.1.

Select the entire picture (using Select All in the Edit menu if you're in SimpleText). Choose Copy from the Edit menu. Switch back to ResEdit, where PICT -16506 should still be open in its own window. Choose Paste from the Edit menu. The MacOS logo will be replaced by your new logo. Save your changes, quit ResEdit, and restart. In place of the normal MacOS logo, you'll see your new and improved logo.

With ResEdit, even a simple task such as copying and pasting can give you some great results. Another simple task—changing resource IDs—can also provide you with some neat Mac tricks.

FKEYs

A good example of how resource IDs can help you customize your Mac is the FKEY resource. FKEYs are small programs stored

If you don't see the changes, you may have edited the wrong picture for your size screen. Alternately, you might have edited the image for the wrong bit depth. Also, if the edited MacOS screen is an image saved in Thousands or Millions of colors, you won't see it at all.

The Buddha's Revelations

If you really mess up a resource file, perhaps by deleting everything in it, you can choose Revert from the File menu to put everything back to its last saved state.

in an FKEY resource in the System that run when you press Command, Shift, and a number from 1 to 0 along the top row of keys. These resource-based programs were named for the function keys that IBM PC keyboards use that Apple tried to avoid for many years, until Apple finally released the Apple Extended keyboards. The most well-known FKEYs included with the MacOS are Command-Shift-3, which takes a screen shot, and Command-Shift-1, which ejects a floppy from an internal floppy drive. While most people have just these FKEYs that come with the System, there are lots of publicly available FKEYs that do various tasks. While you can't actually edit these resources, you can change which number key you use to activate it to make it more convenient or install more FKEYs and organize their Command keys in a better sequence.

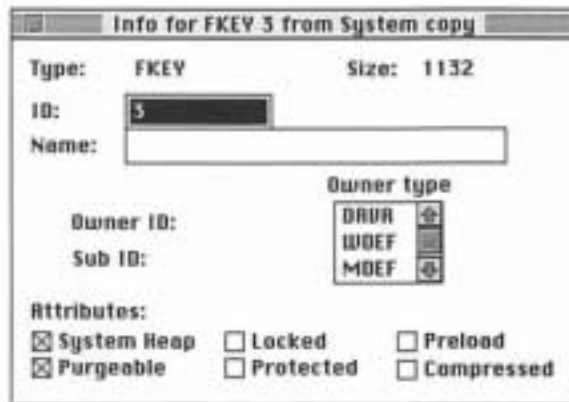
To see how this works, let's change your screen shot FKEY so that you activate it by pressing Command-Shift-5 instead of Command-Shift-3. First, as always, make a copy of your System file and move it out of your System Folder. Open the copy of the System file and open the FKEY picker (see Figure 2.17).



Figure 2.17

The FKEY resource picker from the copy of the System.

Your FKEY picker probably looks a lot like ours, though it might have some extra entries. One of the entries has a resource ID of 3. Select this and choose Get Resource Info from the Resource menu, as you've done before (see Figure 2.18).

**Figure 2.18**

The information window for FKEY resource 3.

When you used this dialog last time, you were interested in the Name field. This time, however, you're interested in the ID field. Change the number in it from 3 to 5. Close the window, save your changes, and quit ResEdit.

As before, switch the System files, saving the unedited version, replacing it with the edited version, and changing its name to just System.

Once you've restarted, press Command-Shift-5. You'll hear the camera click sound, and the Mac will take a screen shot for you (saving it on the top level of your startup disk).

If you get FKEYs from some other source, such as an online service, you'll occasionally want to reorganize them so that the keys that activate them make more sense to you. With a few simple steps, you can change each one to match your needs precisely.

One Resource, One ID

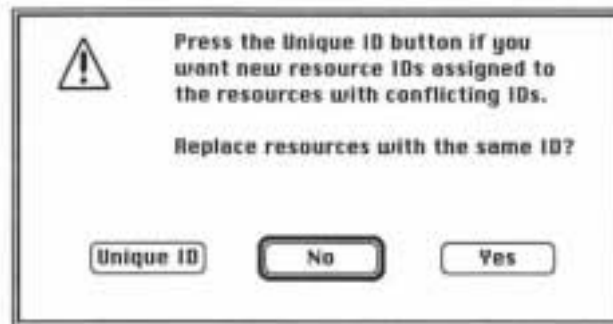
The one exception to changing resource IDs to whatever you want is that each ID needs to be unique within a set of resources of the same type in the same file. ResEdit, therefore, won't let you change the ID of any resource to an ID that is already being used by a resource of the same type in that file. If you want to swap the

IDs of two resources, you'll need to renumber the first one to some other unused number, renumber the second so that it has the ID of the original, and then renumber the original again so that it uses the old ID of the one you just changed.

What happens if you paste in a resource that already has an ID used in that file? ResEdit notifies you that this is happening with a dialog. This dialog enables you to replace the old resource with the new one, assign a unique ID to the new one, or cancel the entire move (see Figure 2.19).

Figure 2.19

When you paste a resource with the same ID as one that already exists in the file, you are presented with the options in this dialog.



Working with Resources Overview

You probably realize that ResEdit works the same as most other Mac applications—resources can be cut and pasted, selected, and edited in the same way you might work with text in a word processor. You've also seen that as long as you use common sense, ResEdit won't cause any more problems than your word processor does while changing some text to a different font.

To create a copy of the System file for editing:

- Locate the System file in the System Folder.
- Select the System file and choose Duplicate from the File menu.
- The Finder creates a new file named System copy.

- Move System copy somewhere else—the desktop is a good spot to keep it.
- Or Option-drag the file to the desktop to duplicate it—however, it will be named System instead of System copy.

To replace the System with the edited copy:

- Locate the System file in your System Folder.
- Drag it out of the System Folder and put it where you can find it in an emergency.
- Drag the edited copy of the System file named System copy (or just System) into the System Folder. Change its name to System if it is not already.
- You need to restart your Mac for any changes to take effect. Please note that if you restart without a System file in your System Folder, your Mac will not be able to start up.
- If your Mac restarts and doesn't seem to have any problems you can't live with, you can throw away the older, unmodified copy of the System file.

To recover from a System file screw-up:

- If you screw up your System and can't start your Mac, refer to Chapter 0 to find out how to recover from this.

To duplicate a resource:

- Use the picker to select the specific resource you want to duplicate.
- Choose Duplicate from the Edit menu (or press Command-D).

To change a resource's name or ID:

- Select the resource you want to modify in the resource picker.

- Choose Get Resource Info from the Resource menu (or press Command-I).
- Edit the ID or Name fields.
- Close the window and accept the changes.

To move a resource from one file to another:

- Open the source file.
- Select the resource in the source file that you wish to copy.
- Choose Copy from the Edit menu (or press Command-C).
- Open the destination file.
- Choose Paste from the Edit menu (or press Command-V). ResEdit puts the resource into the proper picker. If there is an ID conflict with a resource that is already there, ResEdit asks what you wish to do to resolve it.

To create a new resource file:

- Choose New from the File menu (or press Command-N).
- Enter a name for the new file and select a location for the file.

To change the file and creator types of a file:

- Open the file in ResEdit.
- Choose Get Info for [filename] from the File menu.
- Edit the Type and Creator fields.
- Close the window and save the changes.

To replace the contents of a resource (for example, PICT resource):

- Open the source resource in its editor. (In some resources it is necessary to select the contents of the resource by choosing Select All from the Edit menu.)
- Choose Copy from the Edit menu (or press Command-C).

- Open the destination resource (of similar type to the source) in its editor.
- Choose Paste from the Edit menu (or press Command-V).

Safety Tips—What Not to Do!

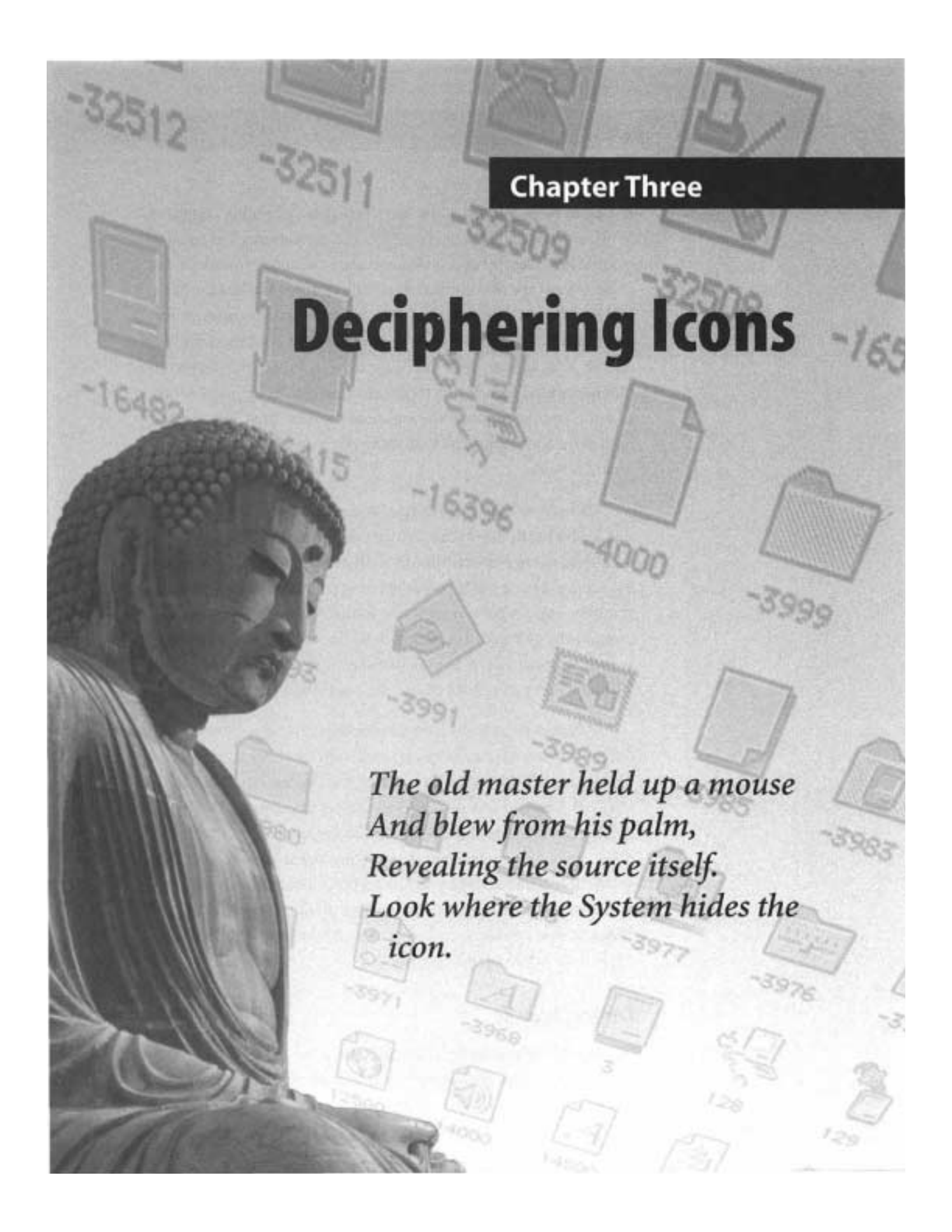
- Again, don't work without a net! Make a complete backup of all your data and system and understand how to restore it in an emergency.
- Read the previous safety tip.
- Don't save your changes if they are wrong!
- Don't *not* save your changes if they are correct!
- Don't restart your Mac without a System file in the System Folder.
- Don't put a System and a Finder in the same folder unless it's the System Folder.
- Don't edit the active System file in the System Folder.

The Buddha's Guide to ResEdit Enlightenment

- Whose creator is ERIC?
- What has neither a type nor a creator?
- What does the Revert command in the File menu do?
- What happens when you hold down Shift-Command-Option-Control and choose About ResEdit?
- How do you get your Mac to display a blinking question mark when it starts up?
- Can you make the Mac oink when you take a screen shot?
- Should you take out all the trash?
- How many sounds can your Mac play simultaneously?
- What is the invisible symbol in 'snd'?

Zen and the Art of Resource Editing

I



Chapter Three

Deciphering Icons

*The old master held up a mouse
And blew from his palm,
Revealing the source itself.
Look where the System hides the
icon.*

It used to be easy to tell a Mac from just about any other computer. When you look at a Mac, you see lots of little pictures that usually make it very clear what it is that you are looking at. Nowadays, just about every type of computer uses little pictures just like the Mac, and they do so for a very good reason. Those little pictures, more commonly known as icons, serve to represent a concept quickly—doesn't it make sense that you delete a file by putting a little picture of it into a little picture of a trash can? Icons in dialog boxes serve to bring your attention to the nature of the files listed. Even your hard drive and floppy disks are shown by icons that uniquely identify them.

These icons, like everything else, are resources. This means that, using ResEdit, you can change them to suit your fancy. You could change your trash can so that it looks like a dumpster, a shredder, a toxic waste container, or any number of other symbols—you may even want it to look like a recycle bin, although why you would is your own business. You can change the icon of your hard drive to look like something more personal. Virtually every icon that you can see can be edited.

Although most icons look similar, there are actually several different types of icon resources, each used for different purposes. Dialogs use ICON and cicon resources; the Finder uses ICN#, ics#, icl4, ics4, icl8, and ics8 resources to display icons on the desktop; and the System uses SICN resources for the small icons that show up in various places (such as the blinking alarm clock icon or the small file icons in an Open file dialog box). Though each of these icons has its own unique attributes, they all use a very similar editor, so you can learn the basics of using the icon editor and apply it to each kind of icon resource.

Dialog Icons

One of the simplest of the icon types is the ICON resource. This resource type is primarily used in dialogs and alert boxes.

You've already seen one of these icons (the warning triangle with the exclamation point) when you were looking at the System for the first time. Now you'll learn how to edit this type of icon so that it looks the way you want. To do so, we'll use another icon that you often see: the stop sign that shows up every now and then to tell you that you really need to pay attention (see Figure 3.1).

Because you'll be editing a resource in the System again, you should make a copy of your System file to edit with ResEdit. Once you've done this, you should open it with ResEdit and you'll see the main System window, showing you icons for all the different resource types in the System. Locate and open the ICON picker to see a list of all the ICON resources in the System (see Figure 3.2).



Figure 3.1

This is what the ICON looks like before you get to it. Heh heh.



Figure 3.2

The ICON resources in your System file.

Double-click ICON 0, the stop sign with a hand in it. This opens that ICON resource in its editor (see Figure 3.3).

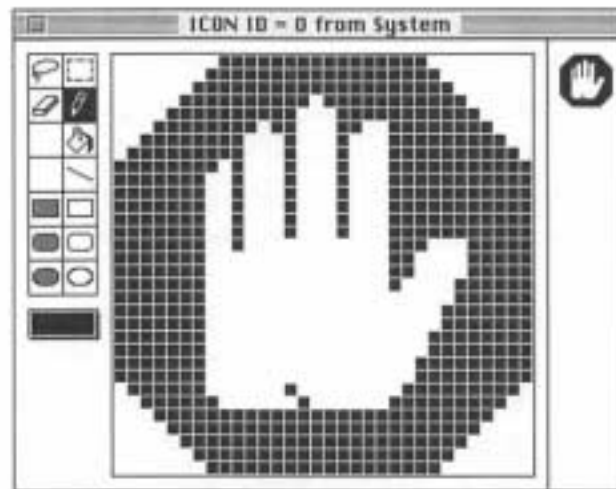


Figure 3.3
ICON 0 opened in its editor.

This editor has two main portions: the editing area and the tools area. The editing area, in the center of the window, is an enlarged version of the ICON picture. Each square you see in the editor is a single pixel in the real icon. You can change these to change the icon, unlike the PICT editor you saw in the last chapter that only let you look. To the right of the editing area is an actual-size version of the icon, so you can see what the icon is going to look like as you edit it.

The tools area of the editor, on the left side of the window, has many of the tools you may be used to seeing in paint programs. The top two tools are used for selecting portions of the picture, with the lasso being for irregularly shaped sections and the selection rectangle for square or rectangular portions of the icon. Below these are the eraser and pencil tools. When the eraser tool is selected, clicking and dragging over sections of the editing area will make them white. The pencil tool turns black squares white and white squares black. This is the tool you use for editing small details of the icon. Below these tools is the paint bucket. When you click the paint bucket in an enclosed area of pixels, it fills that area with the pattern you have selected from the pattern palette at the bottom of the tools area.

Below the paint bucket is the line tool, which enables you to quickly create straight lines at any angle in your icon. The rest of the tools enable you to make shapes, either hollow or filled, with the pattern you've selected.

At the bottom of the tools area is the pattern selection palette. If you click and hold on it, it shows you a palette containing a set of patterns (see Figure 3.4). Still holding down the mouse button, drag to the pattern you wish to use. When you let go, that pattern is selected. If you drag beyond the bounds of the pattern palette and let go, it “tears off” the pattern palette into a floating palette. You can use this for quickly switching between patterns; just click the pattern you want to use in the floating palette.

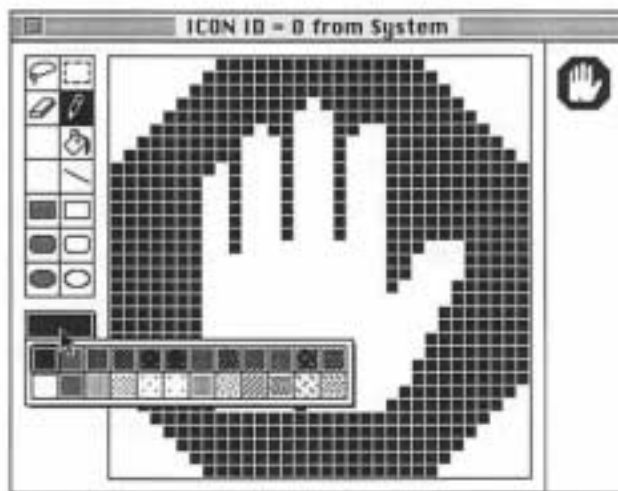


Figure 3.4

ResEdit provides a nice set of patterns you can use to fill in your ICONs.

The ICON editor also adds a Transform menu to ResEdit's menu bar that enables you to alter the portion of the icon you have selected with the selection rectangle or the lasso. Once you've selected a portion of the icon, you can use this menu to rotate that portion, flip it, or nudge it in any direction—you can also use the arrow keys for this (see Figure 3.5 for an example of rotation). You can also use this menu to turn off the gridlines in the editor (try it to see what the icon looks like without gridlines).

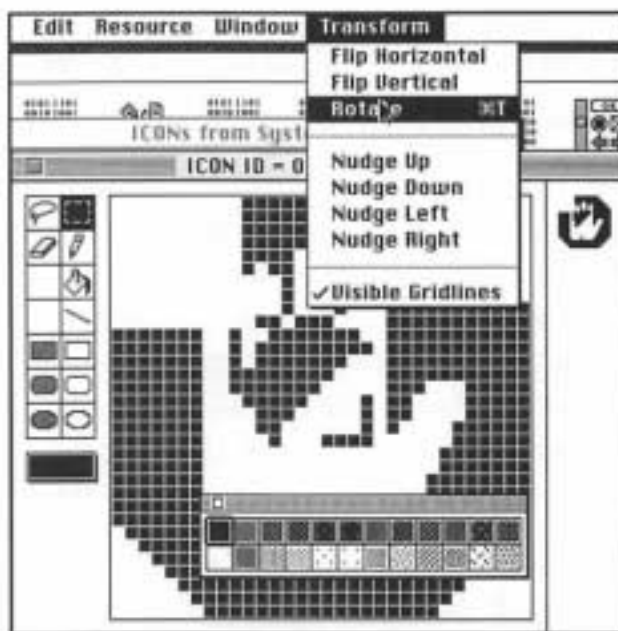


Figure 3.5

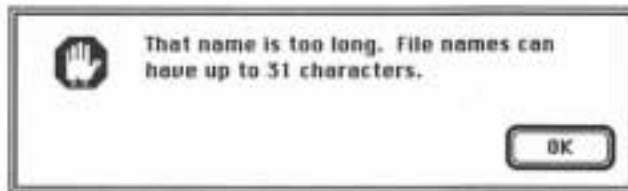
With ResEdit's tools, menus, and palettes, you have the power to make icons as ugly as you want.

Now that you've seen where things are in the editor, try editing the icon. You can design anything you want, of course, but in our example, we used an old trick from grade school (see Figure 3.6). Once you've edited the icon to your satisfaction, save your changes and close the System file. Now move the edited version of the file into your System Folder (you will have to move the version of the file that is already there out before you can do this). Then restart your computer. You do have a backup copy of the System file, right? RIGHT?

To see your changes, you need to get that particular kind of alert dialog to show up on the screen. We found that the easiest way to do this is to edit the name of any file or folder and try to make it longer than 31 characters. Click the name under the icon and just start typing. Eventually, you'll hit the 31-character limit and a dialog will appear. This dialog will have your new icon in it (see Figure 3.7).

**Figure 3.6**

Fortunately, my ICON-editing ability was not graded on aesthetics.

**Figure 3.7**

With the click of a mouse, you too can have silly icons appearing in your dialogs.

cicn Icons

Now that you've edited your icon, your next thought (if you're on a color machine) is probably that you'd like it to be in color. ICONs don't have any color information in them, which is why our changes could only be black-and-white, but there is a way to get color icons into dialogs. All that's required is that you have a cicn ("Color ICoN") resource in the System, and that it has the same ID as the black-and-white ICON version. The System will automatically use this icon in a dialog when your monitor is set in 256-color mode or greater. If you've got a color machine, you can follow along with this example. If you don't have a color machine, you should read through this section anyway, just to help you get more familiar with ResEdit.

In the case of our stop sign icon, the System doesn't already have a cicon with this ID that you can edit, so you'll have to make a new one. Open the copy of your System again (you probably still have a copy somewhere, don't you?), and open the cicon picker. You'll see a single cicon, number - 16396 (this is the color icon that is used in the About This Macintosh window). You need to create a new cicon for your color version of the stop sign icon.

Choose Create New Resource from the Resource menu. It automatically makes a new cicon resource and opens it in its editor. Unfortunately, it makes this new cicon with a default ID of 128, so you need to close the editor (you're not quite ready to edit the icon anyway) and choose Get Resource Info. Change the ID number to 0, which is the same ID that the black-and-white stop sign has, and close the window.

Open the ICON picker from the System window and double-click ICON 0 (the stop sign/turkey). Choose Select All from the Edit menu (or press Command-A), which selects the entire editing area of the ICON editor. Select all of the icon (press Command-A) and choose Copy from the Edit menu (or press Command-C), which takes that graphic and places it on the clipboard, just like copying or pasting a picture. Go back to the cicon picker (the Window menu lets you navigate between open windows), open cicon 0, and choose Paste (see Figure 3.8).

Of course, the icon you copied and pasted is still black and white, so you'll want to color it in. The cicon editor has a couple of features that are different from the ICON editor and make it possible for you to add color. Under the pattern palette, you'll see a color palette. You use it the same way you use the pattern palette, by clicking and dragging to the color you want to use. Notice that there are actually two color palettes, slightly staggered over each other. These affect foreground and background colors (foreground is the one on top) of the patterns in the pattern palette. To

see what these do, select a couple of different colors (one in each layer) and then open the pattern palette. Use the tools to color different areas of the icon—you'll find the bucket tool to be most useful here. Reverse the colors and color the icon again. The foreground color is used in all the areas that are black in the pattern palette of the ICON editor. The background color is used in all the areas that are white in the pattern palette.

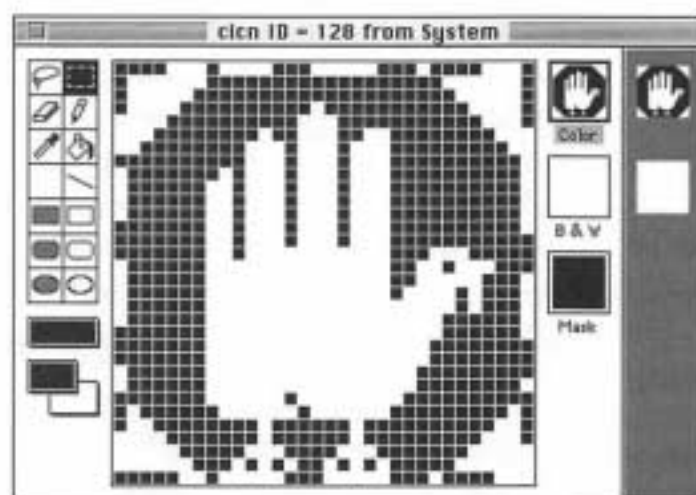


Figure 3.8

It's not color yet, but this is what the cicn looks like right after you've pasted in the artwork from the ICON.

Follow along with us now and open the foreground color palette and choose a red color. Click the paint bucket tool and click a black portion of the stop sign. The sign becomes red (don't forget to change the two black pixels in the hand that don't get filled in by this process). If you want, you can color in the hand portion of the cicn as well and embellish the icon with whatever color you like. (And why not? It's your icon, after all!)

When you're done coloring in the icon, go ahead and save your changes, but don't close the window yet. We'd like you to take a second to look at some of the other features in this editor.

Aside from the color palettes, there is an addition to the tools area. The eyedropper tool, just to the left of the paint bucket,

enables you to “pick up” a color from the icon. Choose the tool and click the hand portion of the icon. The color you clicked on is now selected in the color palette. Now click the stop sign portion of the icon. That red color is selected in the color palette. This is useful if you’re not sure which color you used in one portion of the icon, but you know you want to use the same color in another portion.

Here’s a tip: Click a color with the color dropper. Now Command-click some other color in the icon. Everything that was the color you just clicked gets set to the color you chose with the color dropper.

The final addition to the editor window is in the display area on the right. When you were looking at ICONs, this was just a preview of the black-and-white icon you were drawing. While this editor has, as you might expect, a preview of the color icon, it’s also got a black-and-white version of the icon. At the moment, this is empty, because you haven’t defined it yet. To make a black-and-white version of the icon, drag the icon from the color preview to the square that has B & W under it (the cursor turns into a “grabber hand” as you move over these squares).

You might be wondering why you need a black-and-white version of this icon. After all, won’t the System just use the ICON if you’re on a black-and-white machine? No. Once you’ve defined a cicon with the appropriate resource ID, the System uses that when it displays the dialog (as long as you’ve got a Mac that can do color). If you’re in black-and-white mode, the System uses the black-and-white version of the cicon. And there isn’t one unless you tell ResEdit to make it—which is what you did when you dragged the icon from the color box to the B & W box.

Icon Masks

Under the black-and-white version of the cicon, there’s a square labeled Mask. A mask is information that determines which portions of an icon are transparent and which are not. If a portion of the mask is black, it means that the corresponding portion of the icon is opaque (that is, visible). If part of the mask is white, the

corresponding part of the icon is transparent (not visible). You make a mask the same way you make the black-and-white version of the icon: Just drag from the Color square to the Mask square (see Figure 3.9). By clicking the Mask square, you can edit the mask directly. If your icon is going to appear on a colored background, you can cause some neat effects by editing the mask, such as making window icons that show the desktop underneath. Because the dialog is white, however, you won't notice any neat effects you might try to make, because a transparent portion of the icon over a white background looks the same as a white pixel in the icon. You'll see how to do some interesting things with this when you learn about Finder icons later in this chapter.

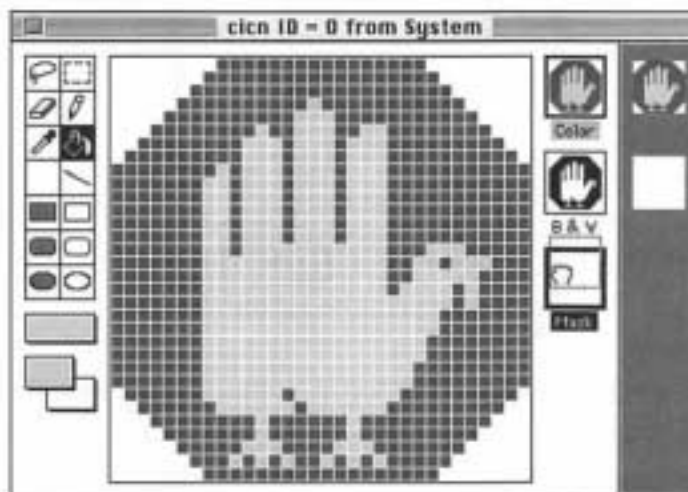


Figure 3.9

Making a mask is easy with ResEdit's slick drag-and-drop mask-making feature.

Once you make the mask, you see the icon show up in the far right portion of the editor. This is a preview area that shows how the icon looks with its mask against a background. By default, this is the standard gray pattern. By playing with the mask and editing it, you can see how the mask affects the transparency of the icon. You can change the pattern used in this preview area by choosing a different pattern (including the desktop pattern) from the cicn menu, which is visible when you're in this editor.

The `cicn` menu, in addition to enabling you to specify the background used in the far right preview area, enables you to change the icon's size. Choosing Icon Size from the `cicn` menu brings up a dialog that enables you to specify the width and height of the icon (see Figure 3.10). The width and height can each be 8, 16, 32, or 64 (they don't have to be the same). Also, you can choose to scale the icon to the new change. This makes for some interesting possibilities, but the dialog needs to be set up to use `cicns` this big, so it won't do you much good to change this yet. You'll see how you can set up dialogs to use big icons in the chapter about dialog resources.

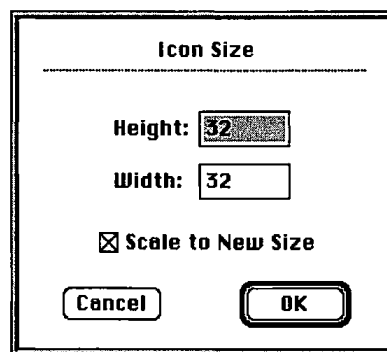


Figure 3.10

Changing the size of the `cicn` through the Icon Size dialog.

The last menu item, Delete B & W Icon, enables you to delete all the black-and-white information from the `cicn` (this is available only when you've clicked the B & W square to bring up the black-and-white editor). In general, you probably won't want to do this; it's just here so programmers can trim some space from their programs if they know the black-and-white information will never be used (for instance, in programs that require color). Doing it yourself could cause some ugly icons to appear when you're in black-and-white mode on your computer.

When you're in the editor for the color version of the `cicn`, the Color menu becomes available. This menu contains a set of commands and options for working with the color palettes at the

bottom of the tools area. The first group of menu items enables you to choose which colors are displayed in the palettes. Apple Icon Colors is a palette of the 34 colors that Apple has specified for icons in its human interface guidelines. If you want your icons to look consistent with the rest of the interface, you should use colors from this palette. Recent Colors is a palette of all the colors you've used recently (that is, while working with this editor). If you want to use a lot of subtle colors, you probably want the next option, Standard 256 Colors. This is the standard 8-bit System palette. Obviously, Standard 16 Colors is the standard 4-bit System color palette and Standard 16 Grays and Standard 4 Grays are the standard palettes used on grayscale screens or when you're in grayscale mode on a color screen.

If you really want a lot of color options and don't care about matching the standard palette or Apple's recommended colors, you can use the last menu item in this group, Color Picker. When this menu item is selected, clicking on the color palette brings up the standard Mac color picker instead of showing you a palette. From the color picker, you can choose any color you want. In general, this probably isn't a good idea, because your colors will get converted to the standard 8-bit palette when used in most applications anyway. When you learn about clut resources in Chapter 11, you'll see how to change the palette from which a program picks its colors.

The Foreground <-> Background menu item lets you swap the color in the foreground color palette with the color in the background palette. This is useful if you decide you like a pattern with the colors you have selected, but want the inverse of what you have currently chosen.

The final menu item in the Color menu is Recolor Using Palette. When you switch to a new palette, you can choose this menu item to have ResEdit pick the colors in the new palette that

are graphically closest to the colors in your icon. To see how this works, try bringing up the palette of the full 256 colors. Choose some dark greens and place a few dots on the stop sign portion of the icon. Switch to the Apple Icon Colors palette and choose Recolor Using Palette. You'll see that ResEdit does its best to match the colors from the full 256 colors with the colors in the Apple Icon Colors palette.

As you can see, ResEdit provides a whole suite of useful tools for editing icons. You can do just about anything with the colors in a particular icon. Remember, however, that when the System displays it in a dialog, it will map the icon to the standard 256-color palette. As a rule, you should use this palette or the Apple Icon Colors palette to make icons that will be used in the System, or else they may not look the way you want.

To see your changes, save the copy of the System file you've been working with, quit ResEdit, replace the old System with your new version, and restart. Bring up the dialog with a stop sign by trying to create a filename longer than 31 characters, and you'll see your new icon!

The Buddha's Revelations

If you want to really have fun with the Apple menu, you might try installing a little utility named Zipple from the CD-ROM. It lets you create an animation that covers your Apple menu icon.

Have some fun and color in all your dialog icons. You can be as creative as you want, or just fill them in with some basic colors. If you come up with something you like, show it to a friend.

Finder Icons

You've seen how to edit the icons that show up in dialog boxes, but there's one place on your Mac full of icons begging to be edited: your desktop. Maybe you've gotten a program that has only a black-and-white Finder icon (which sticks out like a sore thumb on your color machine). Maybe you want your trash can to look like a toxic waste container. And what about custom folder icons? All of these icons fall into a different group of icons known collectively as Finder icons.

You're probably wondering why the Finder needs a special kind of icon. After all, ICONs and cicons seem pretty flexible, right? Actually, they're not flexible enough for the Finder. Consider all the different ways that the Finder needs to display icons: in 8-bit, 4-bit, and black-and-white. In addition, it needs to be able to display miniature versions of these icons in case you're viewing a window by small icon, and it needs to be able to gray them out if they aren't available or they are in use or selected. All these different icons are needed for each Finder icon, and they are grouped together into an *icon family*. Working with an icon family is no different from working with ICONs or cicons—you just have more things to work on.

Icon Families

Though they're called Finder icons, the icon families you see the most in the Finder—folders, the trash can, system icons—are actually stored in the System. Open up a copy of the System, and double-click the icl8 icon in the main window (if you've got a black-and-white computer, you may want to open up the ICN# instead; they're both part of icon families, so it doesn't matter which you choose). This action, as always, opens the picker for that resource. Double-click the empty trash can icon to open this icon in its editor (see Figure 3.11).

This editor looks pretty familiar, because it's the basic icon editor with just a couple of additions for working with whole families of icons. One major change is next to the preview portion of the editor on the right. This portion shows every member of the icon family. icl8s are the larger icons used in 8-bit mode, while ics8s are the small icons used in this mode. Similarly, the icl4 and ics4 resources are used in 4-bit mode. ICN# and ics# resources are used in black-and-white mode. All of the large icons share a single mask, as do all the small icons. To switch between members of the

Why are some of these resource names in all caps and others all in lowercase? In the original Systems, Apple claimed for itself all resource names with all uppercase titles. When System 7 came out, Apple decided that it wanted all the lowercase names for itself too. Because ICN#s predate System 7, they're in uppercase. Everything else was introduced with System 7, so they're in lowercase. Application developers are encouraged to use mixed-case names for their custom resources, because Apple has left them a bit confused.

icon family, simply click the icon you want to edit, and the editor displays it with the correct editor.

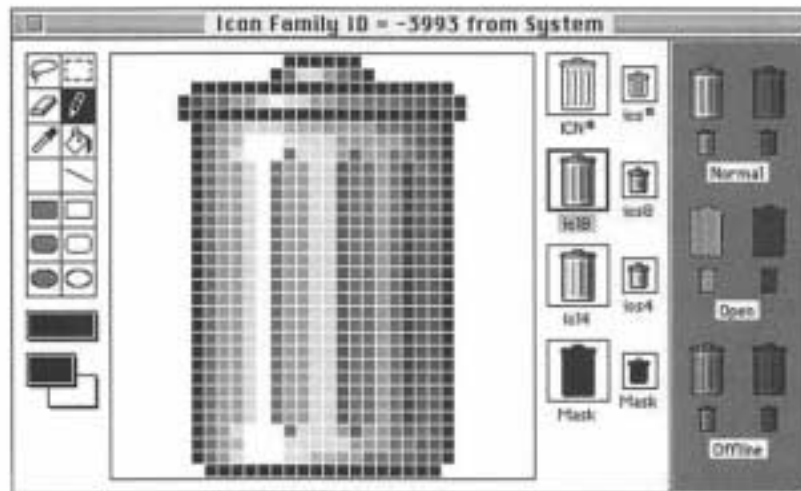


Figure 3.11

The trash icon, opened in its editor with all the other members of its icon family.

Another change in the icon family editor is the preview area on the far right. ResEdit shows you how the selected icon will look (with its small equivalent) in a variety of states. The top row shows how the icon looks, selected and not, when it's just hanging out in the Finder. The next row shows you what the icon will look like when you've opened it. Finally, the last row shows how the icon will look when the volume it's on is not available, as when you have a window open off a floppy and you eject the floppy, or when you are showing a window from a volume mounted across the network and the server is no longer available.

The menus that are available in this editor are similar to the menus that were available in the `cicn` editor. Instead of a `cicn` menu, you have an `Icon` menu that functions the same way (except that you can't change the size of the icon). The `Color` menu is also pretty similar, except there are fewer palette options. This is because the icon needs to conform to the System color palette, which you are limited to in the Finder. Therefore, Apple doesn't even give you the ability to use other color palettes when editing these icons.

Changing the Folder Icon

Now that you've seen what the icon family editor looks like, you can look at a practical example of how you might edit these icons. Close the trash can editor and open the generic folder icon (ID -3999). The `icl8` icon version is selected by default for you to edit. You can make whatever changes you wish—we just colored ours in and changed the tab a bit so that it looks like a manila office folder.

Once you've made this change, you need to make all the other members of the family look similar. To do this, drag the `icl8` square to each of the other squares (see Figure 3.12). This causes ResEdit to do its best job at changing the colors and scaling the icon. If you want to touch up any of these, just click that resource and edit as you want. If you actually change the shape of the icon, you also need to drag it onto the Mask icons so that the mask gets the correct outline.

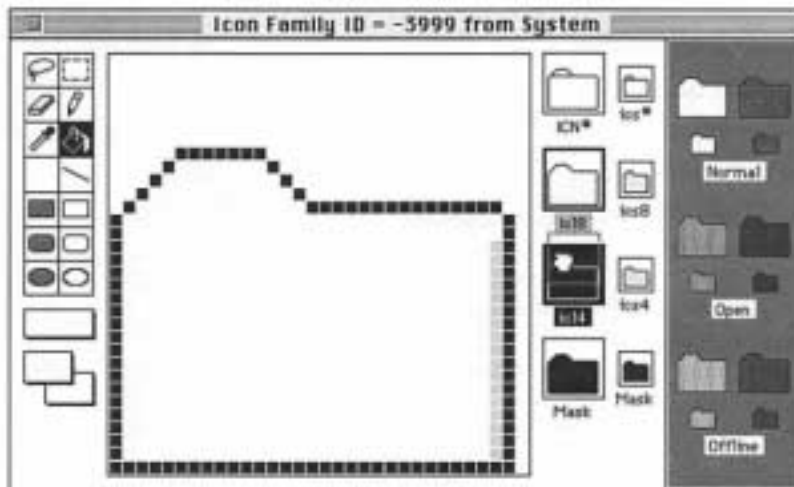


Figure 3.12

Once you've edited the `icl8`, you can quickly make all the other members of the family match.

Once you've got the icon (and all the members of its family) looking the way you want, save your changes to the copy of the System file, replace your System with the edited copy, and restart. When the Finder comes up, you'll see your new folder icon.

Zen and the Art of Resource Editing

Working with icon families is no more difficult than working with the other types of icons; there are just more things to edit. If you want, you can make your desktop look totally customized.

You should keep in mind that the System isn't the only source of icon families; each program you have has icon families for the application and document icons. You can change each of these to make them look like you want. If you do so, however, you'll need to rebuild your desktop (hold down Command and Option as you start up your Mac) so that the Finder knows to go get the new icons for its desktop file.

Another fun way to play with icon families is with System 7's custom icons. You can design a picture and then apply it to anything on the desktop (except the trash can and locked disks) (see Figure 3.13). To do this, go to the Finder and click the hard drive icon once. Choose Get Info from the File menu and click the picture of the icon. It becomes selected. Choose Copy from the Edit menu. Create a new folder on your desktop, select it, and open its info window. Click the folder icon and choose Paste. This places the hard drive icon onto your folder (you can get rid of this by selecting the icon in the Get Info window and choosing Clear).

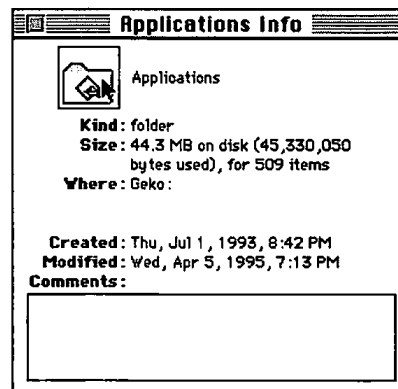


Figure 3.13

Your Macintosh makes it easy to have cool custom icons on your folders and files, and ResEdit makes it easy to touch them up!

When you do this, the System takes the image of the icon, makes all the other members of the icon family based on it, and stores this information in a special invisible file named "Icon " (with a return character after the "n") with the icon family ID number -16455, which is kept inside the folder or volume to which the custom icon was applied (see Figure 3.14).

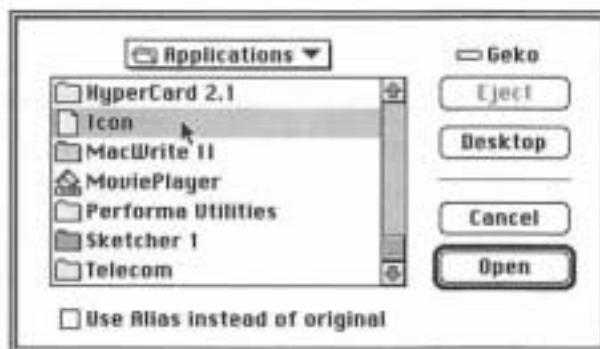


Figure 3.14

The custom icon found on folders is stored in an invisible file named Icon in the folder itself. Fortunately, ResEdit can even open invisible files.

Sometimes the System doesn't do a very good job of creating the other color depth versions of a custom icon, or of setting up the proper mask, or even of creating a decent small icon. You can use ResEdit to go into the file and edit these components of the icon family to touch them up as needed.

Finder icons are some of the most commonly edited resources on the Mac. You can spend hours, if not days, editing the icons you see on your desktop. We encourage you to tweak and edit these to your heart's content, or check on the CD-ROM that comes with this book for tens of thousands of icons that other ResEdit Zennists have already pondered (look in the Icons folder in the Resources folder).

Small Icons (the Other Kind)

You've seen the two main kinds of icons, dialog icons and Finder icons, but there's one additional kind of icon resource: the

SICN resource. SICNs are all over the place, but you don't often notice their presence. The blinking clock that you see when an alarm goes off is one example. The very small icons used in the open file dialog are another example (for example, the little disk icon next to your disk's name). The only really unusual aspect of the SICN resource is that each one is a list of several small icons.

Check out the SICNs in your System file. When you open one in its editor, you'll see all the icons contained in that SICN resource. In those that have more than one icon in the resource, one is usually the actual icon and the other is usually the icon used as the icon's mask. The mask isn't contained in the editor for the SICN, but is simply another SICN in the list. You can switch between them by clicking the icon you wish to edit. The editor is the same as the other black-and-white editors you've seen (see Figure 3.15).

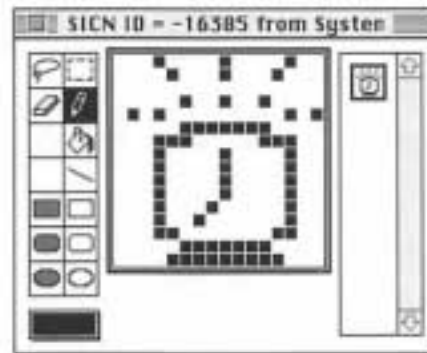


Figure 3.15

The SICN that provides the flashing alarm clock that the Alarm Clock desk accessory flashes, opened in its editor.

Icons Overview

Icons are one of the most visual elements of the Macintosh interface, and also one of the most fun to edit. With a few clicks and a splash of artistic talent, you can make your Mac look really snazzy. Before long, all your friends and coworkers will be oohing and aahing about your creations.

To edit an icon resource:

- Start ResEdit and open the file you want to look at.
- Open the icon picker for the resource you want to edit.
- Double-click the icon you want to change.

To create a mask for an ICN# (or any other member of the icon family):

- Click and hold the mouse button on the icon in the ICN# panel.
- Drag this icon to the panel that says Mask.

To create a new (blank) resource:

- Open up the resource picker of the press you want to add to (for example, ICON).
- Choose Create New Resource from the Resource menu.

To copy the contents of an icon resource:

- Open the source icon in its editor.
- Choose Select All from the Edit menu.
- Choose Copy from the Edit menu.
- Open the destination icon resource in its editor.
- Choose Paste from the Edit menu.

To rebuild your Desktop file:

When editing the Finder icons for an application, you may need to rebuild the Finder's Desktop file (a catalog of icons, types, and creators):

- Hold down the Command and Option keys while starting up until your hard drive is mounted on the desktop and a dialog appears that asks if you want to rebuild the desktop on your hard drive. Note that comments you type into the Get Info window will be lost, but who uses these anyway?

Zen and the Art of Resource Editing

- Click OK to rebuild the desktop for each volume you have mounted.

Create a custom icon using the Finder (in System 7):

- Copy an image (PICT file) to the clipboard.
- Select a document, folder, or unlocked hard drive or floppy.
- Choose Get Info from the File menu.
- Select the icon at the top of the Get Info window (click it or press Tab).
- Choose Paste from the Edit menu.

To edit a System 7 custom icon with ResEdit:

- In ResEdit choose Open from the File menu.
- Go into a folder or volume that has a custom icon applied to it and open the Icon file, or open a file or application that has the custom icon applied to it.
- Select and edit the icl8 (or other family member) with resource ID -16455.
- After saving you may need to restart for the change to take effect.

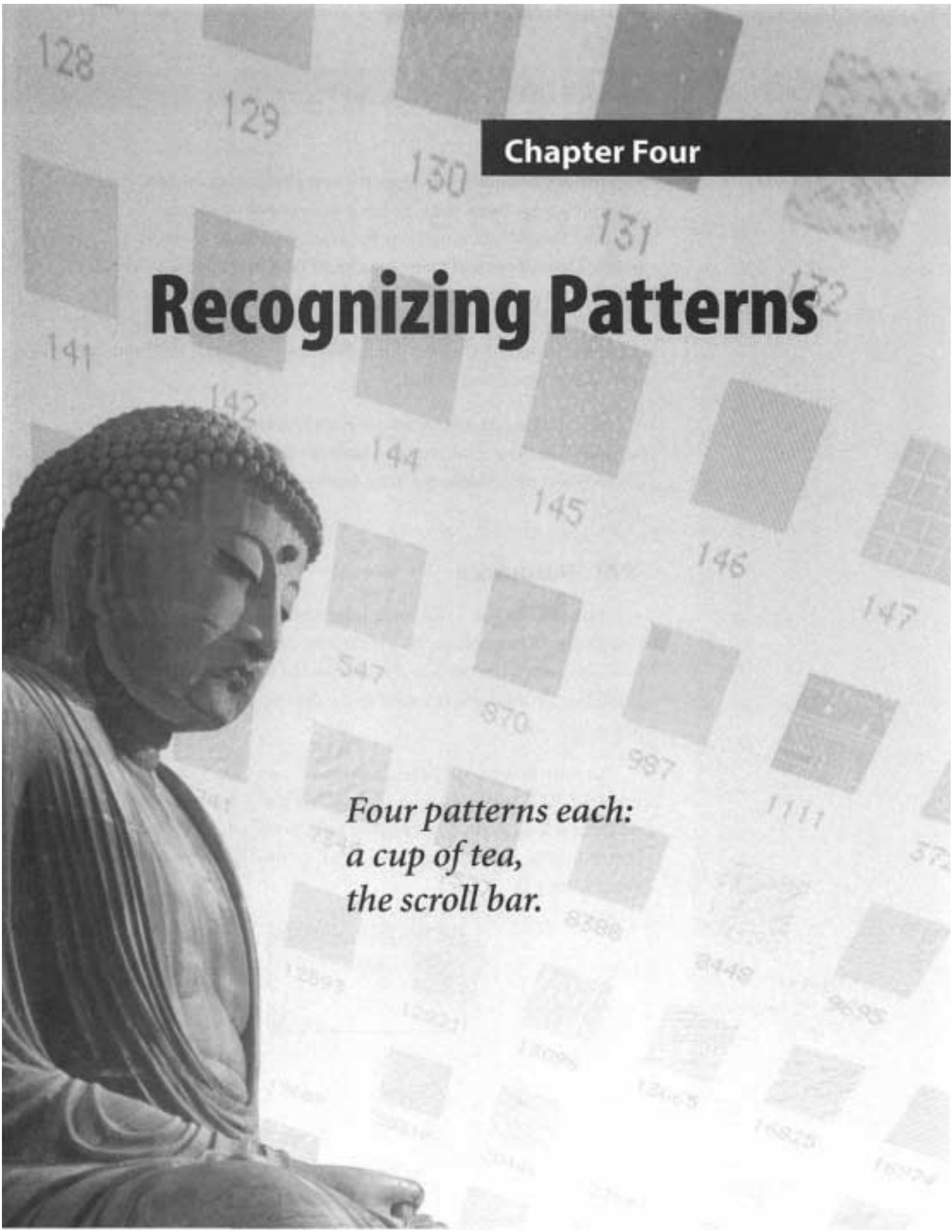
The Buddha's Guide to ResEdit Enlightenment

- What makes a Mac a Mac?
- When you label an icon with a color in the Finder, what controls the application of the tint?
- Can you change the icon in your Welcome to Macintosh dialog box?
- Should you make all your folders look like trash cans?
- When the alarm goes off, who answers the phone?
- Can you make a hole in the middle of an icon that you can't click in the Finder?

- Is the Apple in the menu bar an icon?
- What makes the documents of an application have a particular icon?
- What happens when you hold down the Option key while you're using the pencil tool?
- Going strictly by the book, what is the hardest way to create a new custom icon for a folder?
- Can you make an icon semitransparent?
- Can you remove a folder's icon altogether, leaving just its label?
- Can you make your trash can look really really full?

Zen and the Art of Resource Editing

,



Chapter Four

Recognizing Patterns

*Four patterns each:
a cup of tea,
the scroll bar.*

Patterns are one of the more subtle visual elements of the Mac. They are always there, but very few people notice them consciously. The exception to this is the desktop pattern that most people know about and many even know how to change. However, patterns are present in many other places, such as the scroll bars or the tear-off palette in ResEdit's icon editors. You've changed the scroll bar pattern in Chapter 0, but this time you'll learn what you were doing when you did that.

All of these patterns are simply resources and can be edited with ResEdit. As a result, you can customize the more obvious desktop pattern and also the more subtle scroll bars with very little effort.

'PAT' Resources

The quickest way to do something neat with pattern resources is with the 'PAT' resource. This is the least complex of the various kinds of pattern resources and is used by the System to fill in the scroll bars. In addition, it is used as the desktop pattern on black-and-white Macs.

To learn how to edit patterns, open a copy of the System and open the 'PAT' picker (as with the 'snd' resource, the name of the resource is 'PAT' with a space at the end). Within the 'PAT' picker, you can see 32 × 32 swatches of the 'PAT's stored in the System (see Figure 4.1).

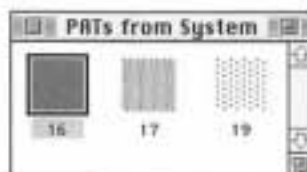


Figure 4.1
The 'PAT' resources from the
System file.

The one we care about is 'PAT' 17. This is the specific resource used by the System when drawing the scroll bar. Double-click the swatch of this pattern to open the pattern in its editor (see Figure 4.2).

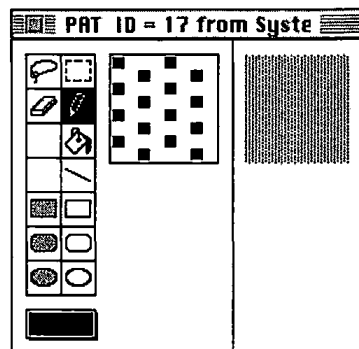


Figure 4.2

'PAT' resource 17 from the System file in the 'PAT' editor.

As you can see, the 'PAT' resource is only eight pixels high and eight pixels wide. The System simply takes this swatch and tiles it through the scroll bars whenever the contents of a window extend beyond its borders. The editor itself is one of ResEdit's standard bitmap editors: The tools are on the left and a sample of what the pattern looks like is on the right.

If you want a better idea of what your pattern will look like when tiled, you can use the Try Pattern menu command in the PAT menu. This takes your pattern swatch and applies it to the desktop. This is especially useful if you're editing 'PAT' 16, which is used for the desktop pattern on black-and-white Macs.

Once you've edited the patterns to your satisfaction, save the copy of the System you've been working in, move the old System out of the System Folder, and replace it with your new System. Restart your machine and you'll see your new patterns in their appropriate places.

ppat Resources: The Secret Behind the Color Desktop

If you are using a computer that is capable of displaying color but is running in black-and-white, ResEdit will tell you that it can't necessarily show you all the colors in a color resource like the ppat, but it will let you edit the resource. And of course you'll be limited to black-and-white ppats.

If you're using System 7.5's Desktop Patterns program or the shareware program Desktop Textures, the swatch on your screen may look like one of ResEdit's generic code icons: a stream of 1s and 0s. This means that you are using a pattern that is larger than ResEdit can display. See the section coming up about larger desktop patterns.

If you've got a black-and-white Mac (one that is not capable of ever displaying color because it doesn't support 32-bit Color QuickDraw), you won't be able to experience the pleasure of ppat resources. If you do have a color machine, you'll find that the ppat resource is a lot more fun than the 'PAT' resource. ppat resources are patterns that can be shown in full color and can be substantially larger than the 8 × 8 tile you are given in the 'PAT' editor. These two features enable you to design a wide range of attractive patterns to use on your desktop. They're one of my favorite resources, as even a nonartist like myself can make something attractive.

Open a copy of your System file, and open the ppat picker. As with the 'PAT' picker, you get to see a swatch of each of the installed ppats. In this case, there is probably only one: ppat 16. Double-click it to open the ppat editor. Notice that this editor is very similar to the 'PAT' editor (and, indeed, to most of ResEdit's bitmap editors), with the obvious addition of the color palette, dropper tool, and Color menu you've seen in the color icon editors (see Figure 4.3).

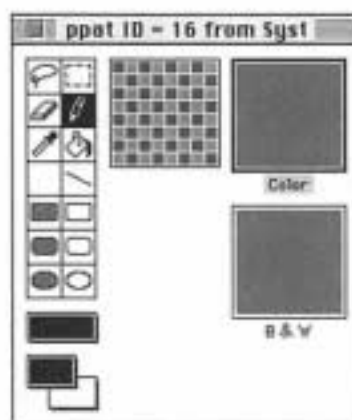


Figure 4.3
ppat resource 16 from the System file in the ppat editor.

In addition to the normal differences between black-and-white and color editors, the editor also has two samples of what the pattern looks like in use. One is the version you will see in color. The other is what you will see if you are in black-and-white mode on a color machine. You can click the black-and-white swatch to edit it to something visually palatable for those times when you're running in black-and-white. Alas, you can't make this black-and-white version of your ppat bigger than 8×8 !

One important difference between the 'PAT' and ppat editors is the Pattern Size command in the ppat menu. When you choose this, you are presented with a dialog which enables you to change the size of the pattern you are editing using any dimension that is a power of 2 (between 8 and 64) for the height and the width. Open this and switch to 64×64 —look in the lower right corner of the dialog (see Figure 4.4). Click Resize to make the change.

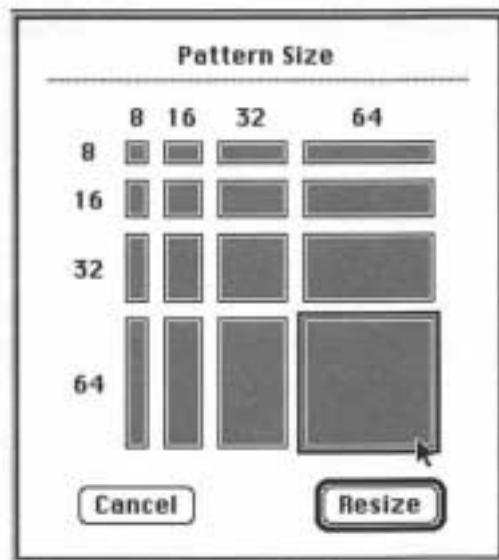


Figure 4.4

The Pattern Size dialog box with 64×64 selected.

Wow! That's pretty big! After you've spent some time making a huge pattern, go ahead and choose Try Pattern from the ppat menu to see what it will look like when spread across your screen

(see Figure 4.5). If you like it, go ahead and save the changes to your new System, replace the older system, and restart.



Figure 4.5
Editing a 64 × 64-pixel
pattern in ResEdit's pattern
editor.

Why do we call the Desktop Patterns control panel a application and not a control panel? Though it does live in the Control Panels folder, it is actually a separate application, as witnessed by its presence in the process menu when it is open. We're just trying to be accurate.

But I Want Bigger Patterns!

Of course, before long, you'll realize that 64 × 64 isn't nearly big enough to make really neat patterns. As it happens, the System can display very large patterns, so the only real obstacle is that ResEdit can't edit them. If you're using System 7.5, the Desktop Patterns application enables you to create very large patterns, as long as you play along with its rules.

The best way to use this application, which doesn't have a built-in editor, is to create your pattern in a paint program and then copy and paste it into the Desktop Patterns window—or, if your paint program supports drag-and-drop via the Drag Manager, you can just drag it from your paint program into the Desktop Patterns window (see Figure 4.6).

**Figure 4.6**

Setting a 128 × 128 pattern with System 7.5's Desktop Patterns utility. This image was created in a ClarisWorks graphic document.

If you do this a few times, however, you'll quickly realize that Desktop Patterns doesn't always retain the integrity of your image. Often, it will resize your image, distorting it in the process. If you want to prevent this, there are a couple guidelines to follow. First, your image's dimensions in pixels need to be exact powers of 2 (that's 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024...). The System can display patterns only with dimensions of powers of 2, so the Desktop Patterns application automatically sizes your image down to the nearest usable dimension. The next guideline is that the total area of your image cannot be greater than 16,384 pixels (or 16K). Thus, if your image has a width of 256, its height can't be more than 64. If you want an image that is 8,192 pixels high, it can be only two pixels wide. If you want the largest possible square that Desktop Patterns will allow, each side needs to be 128 pixels. If you exceed 16,384, Desktop Patterns resizes your image to the closest powers of 2 that give an area of 16,384.

WARNING: If you're using a version of the System prior to System 7.5, there is a slight problem when you use a pattern larger than 8 × 8. If you go into the General Controls control panel, you see that the simple ppat editor in this window shows a black-and-white pattern because it cannot show your large pattern. That's fine, but if you accidentally click this pattern (thereby applying it to the desktop pattern), you will set your background pattern to the black-and-white pattern. That seems reasonable, except that when you applied the pattern to the desktop, you also irrevocably removed your ability to have color patterns (even 8 × 8 ones) as your desktop pattern. To undo this, you will literally need to reinstall your System software, deleting your System file to ensure a clean install.

Zen and the Art of Resource Editing

For those of you who are not yet using System 7.5, the shareware program Desktop Textures does the same thing (except that it doesn't support the Drag Manager).

Bigger still? It's possible, but for that you'll have to use a commercial program such as ScreenScapes, which can provide much larger images (Hans has a 256×256 pattern as his desktop pattern).

And of course, this applies in the other direction as well. Desktop Patterns will enable you to paste in an image as small as 2×2 if you want to use that as your pattern. Funny, we've never met anyone who wanted to do that.

The PAT# and ppt# Resources

In addition to 'PAT' s and ppats, the System contains two other types of pattern resources: PAT#s and ppt#s. These resources are specially designed to hold multiple 'PAT' s and ppats in a list. The PAT# resource is a list of black-and-white patterns, and the ppt# resource is a list of color, 8×8 patterns (see Figure 4.7). In the System, these are the patterns that are available in the General Controls control panel for pre-System 7.5 folks. Thus, you can add your own 8×8 patterns to the list of patterns available in the control panel, making switching between your pattern and another one very easy.

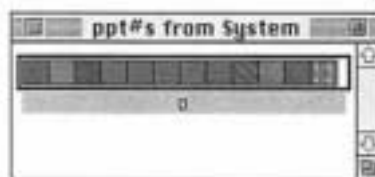
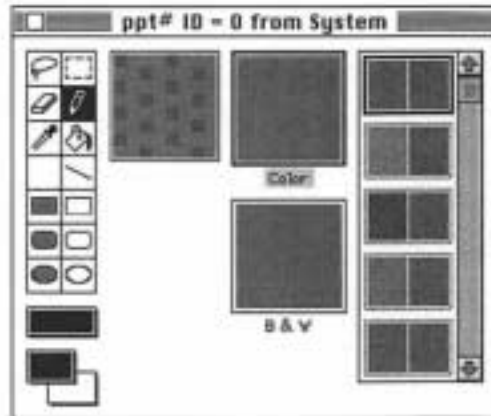


Figure 4.7
The ppt# resource picker.

When you open one of these resources, the scrolling list on the right shows all the pattern swatches within the resource (see Figure 4.8). To edit one of the patterns, click it in the list and edit it to your heart's content (you'll notice that an individual member of the ppt# has a black-and-white component just like its brother, ppat).

**Figure 4.8**

ppt# resource 0 from the System file in the ppt# editor.

If you wish to add a new pattern to the list, simply choose Insert New Pattern from the Resource menu while you're in the editor. And if you want to try any of the patterns you're working on, you can as always choose Try Pattern from the PAT# or ppt# menu (depending on which editor you're in).

ResEdit Patterns

The PAT# resource has another use besides holding the black-and-white patterns that show up in the General Controls control panel. You can use this resource to edit the patterns that ResEdit displays in its pattern palette in any of the bitmap editors. Simply open a copy of ResEdit and open the editor for PAT# 3100. You can add your own patterns or edit its existing ones. Once you've edited ResEdit, save the copy of ResEdit and quit the copy you're currently running. The next time you want to edit a bitmap resource, run the edited copy of ResEdit with your custom patterns.

Please note: If you get a strange dialog box asking if you're going to be using this pattern resource list for desktop patterns, just click Yes. This is a technical option used by programmers.

If you're one of the people who likes to do things that work but that you're not supposed to do, like playing soccer next to a cliff, you can always edit the active copy of ResEdit with itself. You may want to be aware that in order for your changes to take effect, you will have to close the window showing you ResEdit's resources.

Patterns Overview

To edit the scroll bar pattern:

- Open a copy of your System file.
- Open the 'PAT' resource picker by double-clicking.

Zen and the Art of Resource Editing

- Open the 'PAT' 17 resource by double-clicking.
- Use the standard bitmap editing tools.
- Save and switch your System files and restart.

To edit a System 6 and System 7.1 (or earlier) color desktop:

- You must be using a color-capable Macintosh.
- Open a copy of your System file.
- Open the ppat resource picker.
- Open the ppat 16 resource.
- If desired, change the size of the pattern by choosing Pattern Size from the ppat menu.
- If desired, try the pattern by choosing Try Pattern from the ppat menu.

To create desktop patterns larger than 64 × 64 with System 7.5:

- Copy an image to the clipboard from any graphics software.
- Paste into the Desktop Patterns application.
- To keep your image from being resized by Desktop Patterns, make its dimensions in even powers of 2 with the area equal to or less than 16K.

To change the list of patterns in the General Controls control panel (pre-System 7.5 only):

- Open a copy of the System with ResEdit.
- Edit the PAT# 0 resource (or the ppt# 0 resource if you're on a color machine).
- Replace the System file in your System Folder with your copy and restart.

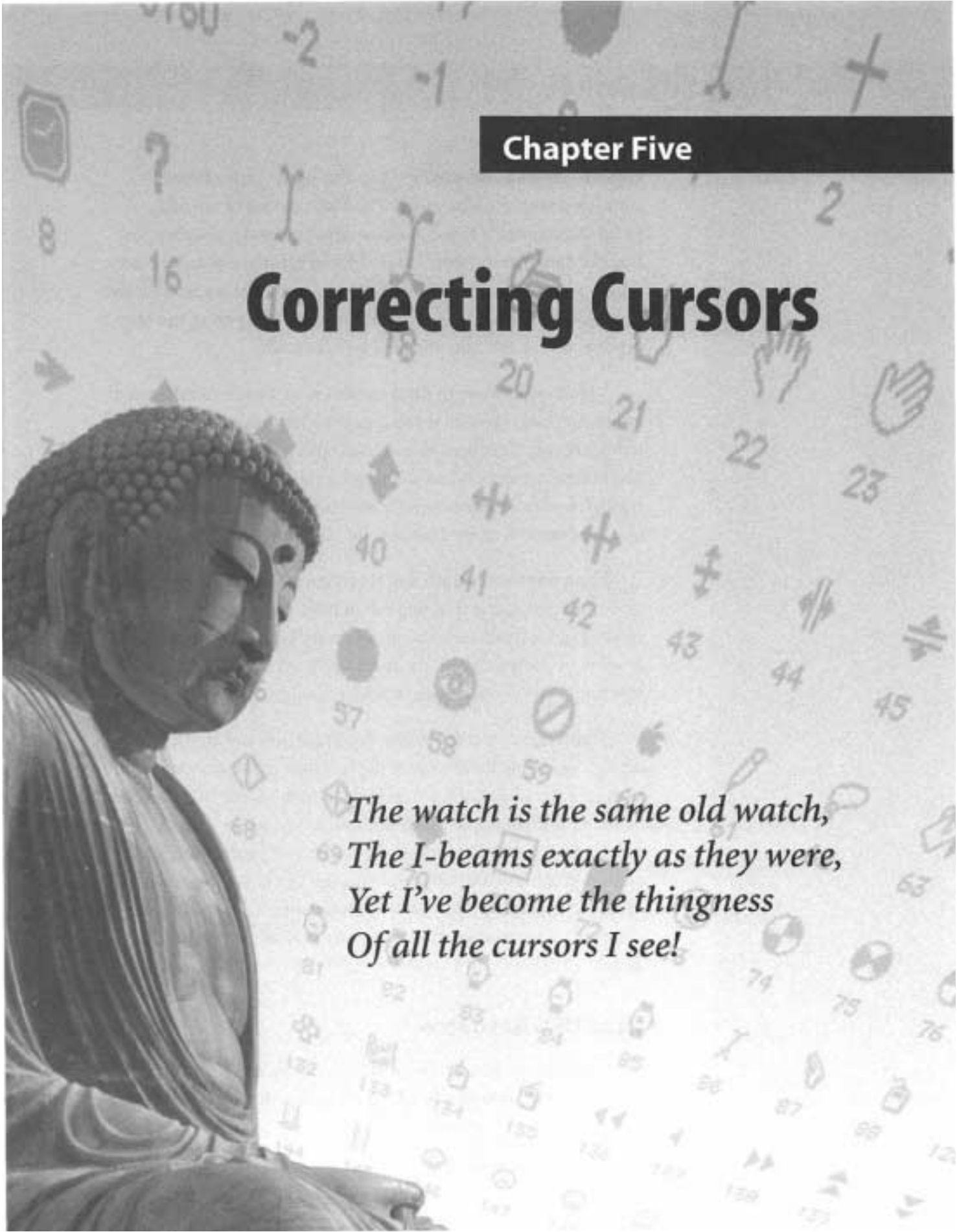
To change the patterns in ResEdit:

- Duplicate ResEdit and open the copy.
- Edit the PAT# 3100 resource.
- Save and open the modified copy of ResEdit.
- To see them, open any graphics editor, such as icons, and pull out the pattern palette.

The Buddha's Guide to ResEdit Enlightenment

- What does 'PAT' 19 in the System do?
- How do the cool large gradient patterns in System 7.5's Desktop Patterns work?
- If you have a pattern 256 pixels wide, what is the tallest it can be for the System to be able to display it?
- Ponder the ugliest pattern (perhaps a desktop of hand-turkeys).
- How many patterns can there be in a ppt# or PAT# resource?
- What makes something a control panel?
- What's the difference between a quilt and a pattern?
- Presuming Peter Piper picked a perfect pixel pattern, how many ppats would Peter Piper possess in his ppat picker?
- Do you see a ppattern here?

Zen and the Art of Resource Editing



Chapter Five

Correcting Cursors

*The watch is the same old watch,
The I-beams exactly as they were,
Yet I've become the thingness
Of all the cursors I see!*

Cursors—arrows, I-beams, and big plus signs—are intimately connected with the Macintosh. The Mac was one of the first personal computers to use a mouse, and cursors let people know how they are moving their mouse by giving them a visual connection to their physical movements. The cursors that are familiar to us all (especially the arrow cursor) are as much a part of the Mac experience as trash cans and smiling Mac icons.

Unfortunately, one of these cursors is an all-too-familiar part of the interface. As soon as you give the Mac a task that takes more than a second, it changes your reliable arrow into a watch cursor and makes you stop. When things get longer than that, the Mac tries to mollify you by switching the static watch cursor to an animated version of the cursor, spinning the minute hand around.

It's an unpleasant truth that watch cursors (or their equivalents in various programs: spinning beach balls, counting fingers, or rotating globes) will never be unnecessary on a Macintosh. The faster your computer gets, the more you'll try to do. Try to do too much, and you're once again watching an animated watch.

ResEdit gives you the ability to edit cursors and animated cursors, changing them to your liking. Some of the cursors, such as the pointer and watch, are embedded in the Mac's ROM and are therefore unchangeable. However, animated cursors such as the watch cursor in the Finder and the "beach ball" cursor used in many applications are easily editable. You can make your own animated cursors so that you can at least watch something you designed as you wait for your computer to finish whatever it's doing.

The CURS Resource

The resource that defines what a cursor looks like is the CURS resource. To see how to edit CURS resources, duplicate and open a

copy of your Finder—the Finder is within your System Folder and we aren't using the System file this time—with ResEdit and open its CURS resources in the picker (see Figure 5.1).

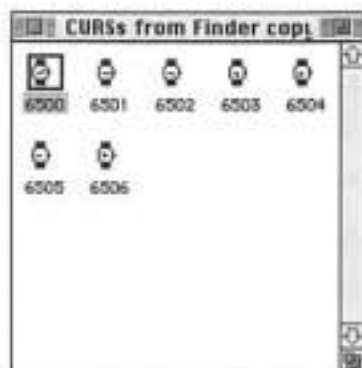


Figure 5.1
The Finder's CURS resources.

You can see that the Finder contains all the animations for its particular watch cursor. Frequently, programs such as the Finder define their own cursors instead of getting them from the System. To see the CURS editor, double-click CURS 6500. You'll see that this is yet another one of ResEdit's familiar bitmap editors, with a few features specific to editing cursors (see Figure 5.2).

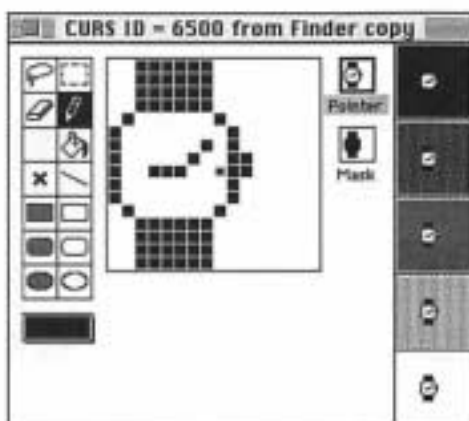


Figure 5.2
CURS 6500 in the CURS editor.

The Hot Spot

The first change is a new tool in the palette, the hot spot tool. To understand what this new tool does, think of the way a cursor,

such as the arrow, actually works. If you were to click in the close box of a window with the stem of the arrow (that is, the lower right portion of the cursor), the window wouldn't close. You have to click the close box of the window with the very tip of the arrow. For every cursor, there is one exact point that actually causes the computer to notice a click in a certain place. This is the cursor's hot spot. The rest of the cursor is simply a graphic to make it obvious where the cursor is on the screen and where its hot spot is.

The hot spot on the cursor is also indicated by a small "x" in the editor. (For the watch, this is initially near the 3 o'clock position.) To change the hot spot, move the hot spot tool (which also looks like an "x") to the new location and click. You will see that the hot spot is at the location you just clicked.

There are some guidelines about placing hot spots that you might want to keep in mind. If you're designing a cursor, you probably want to put the hot spot in some intuitive place on the cursor. For instance, you might choose to put it at the tip of a pointing image or in the center of some radial cursor. Either way, it should be obvious to the user where the active portion of the cursor is (see Figure 5.3).

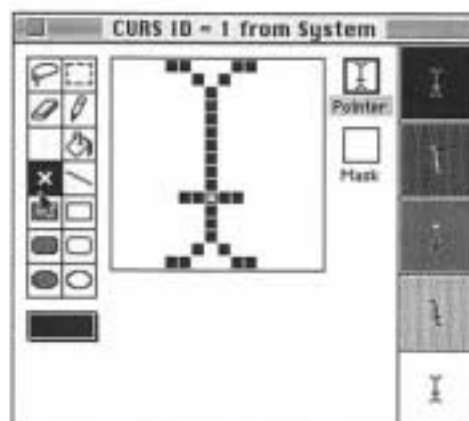


Figure 5.3
Setting the hot spot of a
cursor in a logical position.

If you are creating or modifying animated cursors, which are made up of a series of cursor images, you should keep the hot spot in the same place in every frame. In the animated watch cursor, for example, it would be strange, although no doubt interesting, to put the hot spot of each frame at the end of the changing minute hand. This would cause the hot spot to appear to orbit around as it animates, confusing the user. But we'll explore animated cursors more later.

Cursor Masks

The CURS editor also shows you how the cursor looks on a variety of backgrounds and gives you the opportunity to edit the cursor's mask, just like the ICN# resource.

Masks of cursors don't act quite the same as masks of icons, however. With cursors, any pixel that is not masked doesn't disappear, but instead reverses its color in relation to the pixels behind it. For instance, if a cursor is in front of a white background, it is black, and if it's in front of a black background, it is white—unless it is masked (see Figure 5.4). A common example of a cursor that normally reverses is the text-editing I-beam cursor in the Finder and most applications.

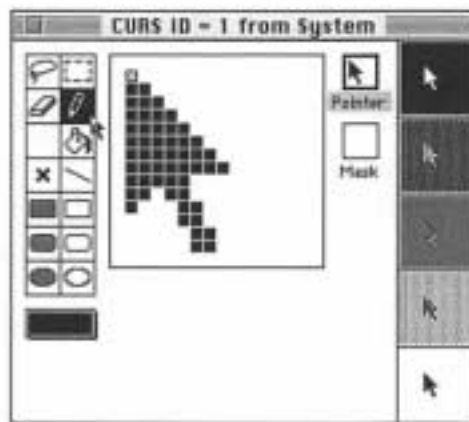


Figure 5.4

The standard arrow cursor without a mask. Notice how odd this cursor looks on the various background patterns.

Masking a cursor can be very useful for helping with its visibility. Consider the Mac's standard black arrow. When it moves over a black background, it doesn't become invisible because its mask creates a white outline. The mask for the arrow is one pixel wider than the actual arrow, and every pixel in the mask that does not have a corresponding pixel in the cursor image looks white.

If you want a better idea of how your cursor looks and behaves, you can choose Try Pointer from the CURS menu, which is visible when you are in the CURS editor. You can move the edited cursor around the screen and look at it against your desktop, or on top of your icons, or whatever. You can click in the bitmap editor (when inside the editor proper, the cursor reverts to the tool you last selected) to change the cursor while continuing to "try" it—it updates your changes on the fly. Of course, if you click in any window other than the CURS editor, the cursor switches back to the normal arrow. However, bringing the CURS window forward again (assuming you haven't closed it but have simply made another one active in the meantime) causes your cursor to once again look like the one you are editing.

Color Cursors?

The Buddha's Revelations

A fun utility on the CD-ROM is named *Color Cursor*. It lets you have a color arrow cursor.

Because it patches the arrow cursor itself, you can edit the cursors in this utility to create your own neat arrows.

They're tough to find, but some programs take advantage of the *crsr* resource to display colorful cursors. You can edit these if the program has them, but you can't force the Finder to use a color watch cursor in the same way you can get the System to use a *cicn* if one exists. You're just stuck with the fact that unless you use some third-party utility to give yourself color cursors, such as *Color Cursor*, which is on the CD-ROM, you'll see a black-and-white cursor when in the Finder or other programs.

The editor for the *crsr* is, predictably, just like the editor for CURS resources, except that it adds the by-now-familiar color tools and menus. Although there aren't any *crsrs* in the System,

you can see the editor by looking at the `crsr` resource in Color Cursor (this is also how you could design your own color cursor, if you choose to run this extension).

Animated Cursors

Of course, `CURS` and even `crsr` resources are only individual frames. Animated cursors consist of a series of `CURS` (or `crsr`) resources. To actually animate the cursor, the Finder (and every other program) refers to an `acur` resource. This resource controls all the aspects of the cursor's animation, from how many frames there are in the animation, to how much time to wait between each frame, to the order in which the frames appear.

The `acur` Editor

To explore an `acur` resource, open the Finder's `acur` picker. You should see only one `acur` resource, number 6500. Open this to display its editor (see Figure 5.5).

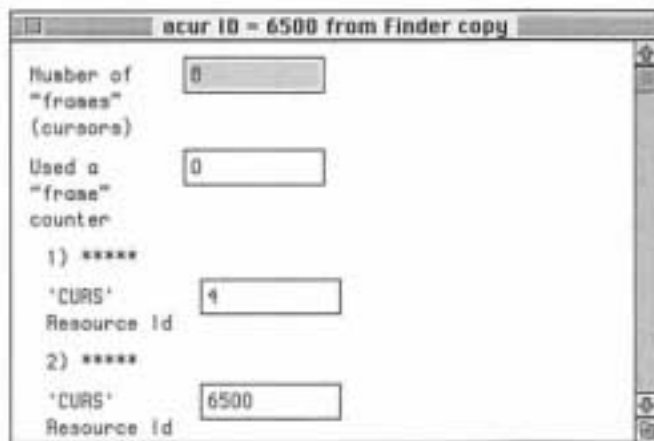


Figure 5.5

The `acur` editor displaying `acur` resource 6500 from the Finder.

This list resource type editor isn't nearly as pretty a resource editor as the bitmap editors we've been exploring up until now in ResEdit, but it's actually very common. Basically, you just fill in the

fields to edit this type of resource. Because each of the fields is labeled, it's usually pretty easy to tell what a particular value is for.

The first field in the `acur` editor tells the program how many cursors make up the animated cursor. As you can see, there are eight frames that make up the Finder's animated watch.

The next field enables you to control how fast the cursor animates. The number in this field is the number of "ticks" (each of which is 1/60th of a second) between frames. If you want one second between each frame, you'd put a 60 in this field. Most of the time, this number is left at 0, but you may want to slow down your cursor for effect.

The remaining fields in this editor describe the order of the frames in the animation. Each one contains the resource ID of a `CURS` resource. First, you might have noticed while we've been exploring the Finder's animated watch cursor, the `acur` resource shows that there are eight frames in the animation while there are only seven cursor resources in the Finder. With a little investigation, though, you'll realize that the first `CURS` resource listed in the Finder's watch cursor `acur` resource doesn't actually refer to a `CURS` resource in the Finder. With yet more investigation, you may realize that it is pointing to the System's resource fork. Because programs have access to the System's resources, the Finder can use `CURS` ID 4 to get one frame of its watch cursor.

The remaining fields in the `acur` list the IDs of `CURS` resources in the Finder. If you look at each of these `CURS` resources in the `CURS` picker, you will see that the sequence looks like a watch with its hand turning—the watch cursor!

Adding to an Animation

If you want to insert a new field in a list resource between two existing fields (or after the last field), click the row of asterisks

before the field that you wish to add the new field after to select it (see Figure 5.6). Choose Insert New Field(s) from the Resource menu, and it adds a new field to the list.

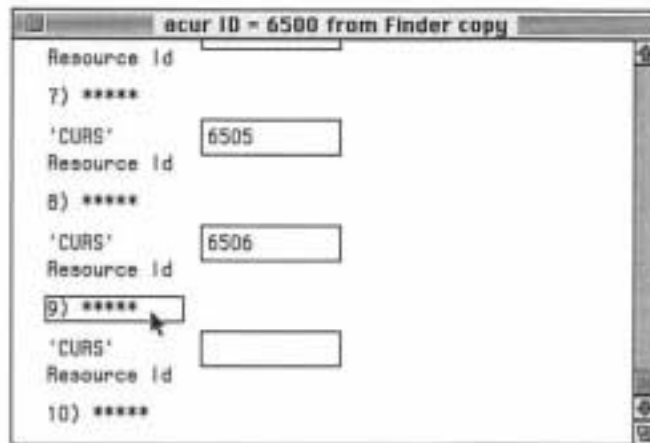


Figure 5.6

Selecting the asterisks to add or delete a field from a list resource.

Obviously, if you do this, you need to change some things about the animation. First, you need to adjust the first field in the acur resource, the number of frames. After all, you just added one. Next, you need to make a new CURS resource with the ID you just typed into the field. To do that, go into the CURS picker and choose Create New Resource from the Resource menu. When the editor appears, choose Get Resource Info from the Resource menu, and change the ID to match the one you just typed into the field. Obviously, you can also choose to make the CURS resource and then insert the ID of the new resources into the acur resource.

You and Your Arrow

Armed with this newfound knowledge about editing cursors, you're probably totally gung-ho to go turn your arrow cursor into something really neat. One problem: The arrow cursor isn't in the System (or if it is, it was put there by some other program). Most programs use the arrow cursor tucked away in the ROMs of your computer. If you absolutely want a different arrow, see the next

section for third-party utilities that enable you to have a different arrow cursor.

You can change the animated cursors, but those changes are in effect only when you're in the particular program that uses them. Even the one in the System isn't used by every application. When you switch to another program, you're stuck with the cursor used by that program, unless you wish to take the time to edit that one as well.

So why edit cursors, you might ask? If you're creating a HyperCard stack, you can use custom cursors to enhance the interface. Or if you're using an older PowerBook with a slower passive-matrix LCD display, you might want to thicken the I-beam cursor in your favorite word processor so that it doesn't "submarine" away every time you move it.

Cursor Utilities

If you do want to do things such as change your arrow pointer or globally change your watch cursor, you're going to need something other than ResEdit (although you can use ResEdit to design the cursors). There are a number of utilities that patch the System and enable you to do these things. Unfortunately for the extension-phobes, they all require some extension or another to properly patch the System. One of these, CursorAnimator (which is on the CD-ROM), is a shareware program that just helps you work with cursors. Another package, ClickChange from Dublin-Click Software, enables you to edit many things in addition to cursors. I have always liked CursorAnimator more because I don't trust extensions that change too many things at once.

Another valuable utility is a better acur editor for ResEdit. ResEdit enables third-party programmers to develop editors, and Chris Reed has written one that far outdoes the boring acur editor

you've seen in this chapter (see Figure 5.7). It's included on the CD-ROM (in the Chris Reed folder in Editors and Templates folder), and if you want to install it, you can see Chapter 15 about installing custom editors.

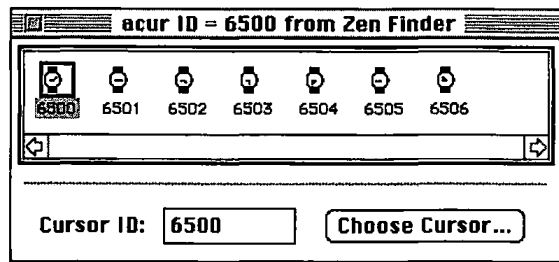


Figure 5.7

Chris Reed's acur editor installed into ResEdit, displaying the acur resource from the Finder.

As you can see, cursors provide a tremendous opportunity for personalizing your Macintosh. Your watch could be replaced with a long, complex sequence such as the cursor on the CD-ROM featuring the story of Robin Hood (though hopefully you'll never have to wait so long that you see the entire sequence) or a simple animation such as sand through an hourglass (these are the days of our lives?).

Cursors Overview

Cursors are a fact of Macintosh life, so you might as well make them interesting. With some judicious application of the CURS and acur editors, you can create cursors that liven up your screen and keep you entertained while you wait.

To edit the Finder's watch cursor animation:

- Duplicate and open a copy of the Finder.
- Open the CURS resource picker from the Finder.
- Select each frame you wish to change and open it in the CURS editor.
- Set the hot spot consistently with your other edits.
- Close the CURS resource editors and picker.

Zen and the Art of Resource Editing

- Open the acur resource picker from the Finder and open acur 6500.
- Set the number of frames of your animation.
- Set the CURS Resource ID fields to the appropriate IDs for your animation.
- You may want to set a delay time between each frame of your animation to slow it down by changing the field labeled Used a “frame” counter.
- You can add additional CURS references to your acur by selecting the row of asterisks between the fields you want to insert between and choosing Insert New Field from the Resource Menu.

To edit a cursor:

- Open the application containing the cursor you want to edit.
- Open the CURS resource picker.
- Open the CURS resource in its editor.

To change a cursor’s hot spot:

- Open the CURS editor for the cursor whose hot spot you wish to change.
- Select the hot spot tool (it looks like an “x”).
- Click anywhere in the editing area to set the hot spot to that pixel.

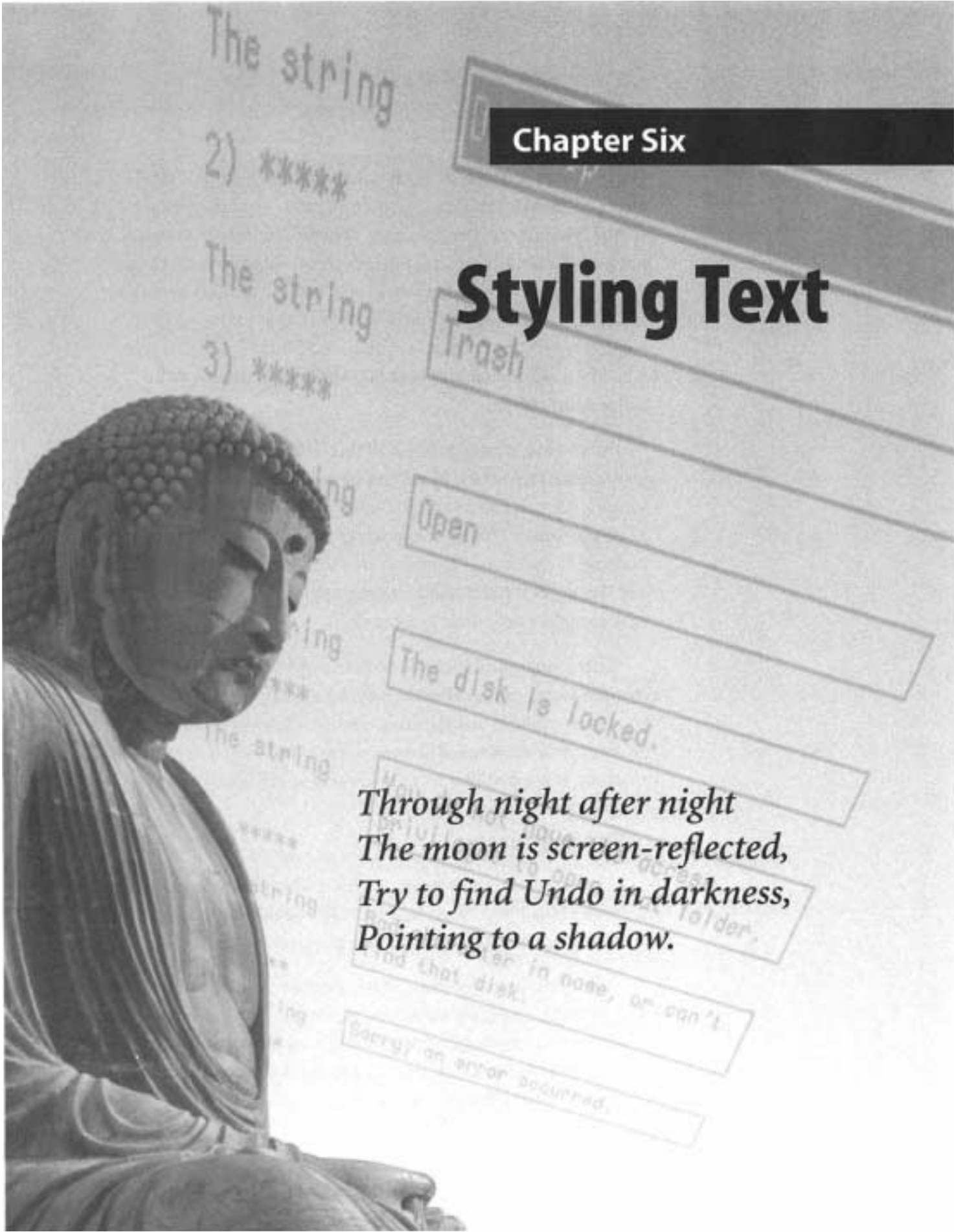
To mask a cursor:

- Open the CURS editor.
- Click the Pointer panel and drag down to the Mask.
- To edit the mask, click the Mask panel.

The Buddha's Guide to ResEdit Enlightenment

- Which cursors are opaque and which are transparent?
- Can you make a left-handed arrow cursor?
- Who came from Locksley?
- What enhancements can you make to HyperCard's hand cursor?

Zen and the Art of Resource Editing



Chapter Six

Styling Text

*Through night after night
The moon is screen-reflected,
Try to find Undo in darkness,
Pointing to a shadow.*

The resources you've seen so far have represented graphic elements of the Mac interface. Just as important—and ubiquitous—are the various text elements of the System. Your trash can doesn't just *look* like a trash can, it says Trash right underneath it. When a dialog appears, it might have an icon in it, but it also has some text telling you why it's appearing. Even the words at the top of your windows in the Finder that say you have 21 “items” and that there is all of 1.2 MB available of your 500 MB hard drive have to be defined somewhere.

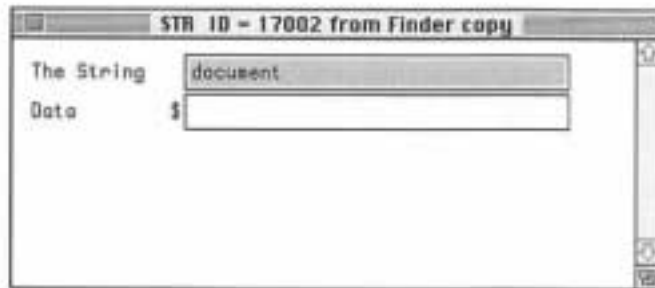
As you might have guessed, all this text is stored in a set of text-oriented resources. By editing some of these specific resources, you can make your Mac label things differently than it normally would. This is yet another good example of the basic purpose of resources—to customize the Mac for different cultures and languages throughout the world, Apple's System programmers have to replace only these text resources.

This swings the door wide open to some interesting and fun changes to you can easily make on your Mac. If you've changed your trash can icon to something less household-oriented such as our favorite toxic waste container, you may not want it to say Trash anymore, but something more in keeping with the new theme.

'STR ' Resources

The simplest type of string resource, which is a resource that contains up to 256 characters, is the 'STR ' resource (once again, this resource's name contains a space at the end).

Lots of programs contain 'STR ' resources, and it's often fun to look through a program's resources to see what sorts of error messages or interesting phrases are scattered throughout. If you open 'STR ' resource 17002 in a copy of the Finder, you'll see that it contains the word “document” (see Figure 6.1).

**Figure 6.1**

'STR' resource 17002 from the Finder.

Obviously the word "document" is used throughout the Macintosh. But where is this particular resource used? If you are in a list view in the Finder and it can't find the application that created a file, it calls it a document in the Kind column of the list view. The Finder gets the word "document" from 'STR' 17002 (a fact we discovered through trial and error, the same way you'll often stumble across stuff). You can change this to be something else such as "generic document" or "stupid computer doesn't know what this file is" or even "orphan"; the only limitation is that an 'STR' resource's text can't be more than 256 characters long. You can then replace your Finder with the one you've edited and all files without parent applications will now have "generic document" or whatever you changed it to in the Kind column.

If you don't have a Kind column in your list views, you may have told the Finder not to display it via the Views control panel. To show it again, open the Views control panel and check the Show kind box in the lower portion of the window.

STR# Resources

Another resource used to hold strings is the STR# resource. What's the difference? Programs use STR# resources differently from 'STR' resources, because STR#s are lists of strings. When using an 'STR' resource, a program just grabs the ID it wants and puts that string where it needs to. With an STR# resource, the program still has to call it by ID, but it has to specify which

Zen and the Art of Resource Editing

element of the list it wants. You won't have to worry about this difference if you're customizing your Mac, except to realize that an STR# editor actually has several different strings in it, not just one.

Actually, whenever you see a # sign in a resource name, it means that the resource contains a list of items. Even an ICN# contains the actual icon and the mask.

For instance, open STR# resource 11750 from a copy of the Finder (see Figure 6.2).

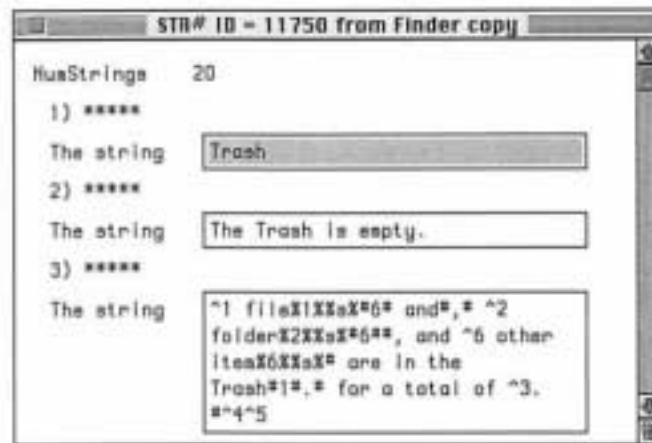


Figure 6.2

STR# resource 11750 from the Finder.

The first item in this list, Trash, is used by the Finder as the text to display under the trash can. If you want to change it, just type the new name in the field. This STR# resource contains several messages about the trash, so you can change these to reflect the new name of your trash can. (In Chapter 9 when we get to menus we'll even show you how to adjust the Empty the Trash menu item to reflect your new theme.)

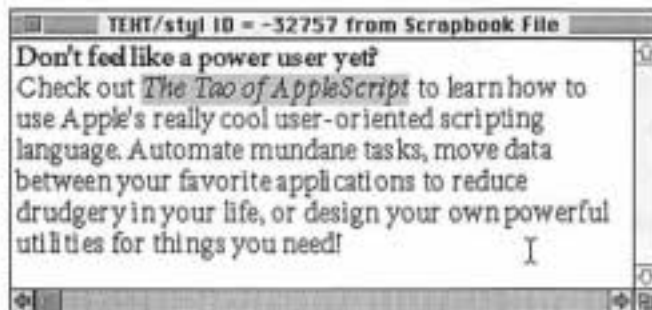
As with 'STR' resources, the best way to find the interesting snippets of text throughout the interface is to explore. Almost every program uses 'STR' s or STR#s, so it's worth looking through your favorite programs to find out what you can.

TEXT and Styl Resources

The 'STR' 's and STR#s can't service every need for text. They're of limited size, and the string isn't stored with any sort of formatting information, so the program has to provide that when it displays it. However, the Mac operating system provides two resources for letting programs store more complex text: TEXT and styl. TEXT resources contain actual text and are combined with a styl resource containing all the formatting information. When you open either of them in ResEdit, they look the same because the editor marries the two resources and enables you to edit the text and the formatting all in one view (wow, just like it should!).

The editor looks and acts just like a minimalist text editor such as SimpleText. When you're in this editor, you get Font, Size, and Style menus added to your menu bar.

Not many programs use these resources, so it might take some time to find them. One place to find them is in the Scrapbook File in the System Folder, provided you've pasted some text into the Scrapbook. Why would you want to edit this? You might put some text you intend to use frequently into your Scrapbook, and then realize that you mistyped it. Rather than open a word processor, copy the text from the Scrapbook, paste it into the word processor, edit it, copy it, paste it into the Scrapbook, and then delete the original, you might want to just use ResEdit to edit the text directly (I know I would!) (see Figure 6.3).



If you see any text that contains things like ^0 or %1% or some variant on these, this means that the program fills in the text on the fly, depending on the situation. For instance, the 15th string in STR# 11750 is the generic error message telling you that a particular item can't be deleted. The ^2 and ^1 are placeholders that are filled in with, respectively, the name of the item and the reason. Of course, some of the text that is put into the placeholder is stored elsewhere in the file, such as the error messages that describe why the item can't be deleted.

Also, don't delete those special characters, because the program won't be able to use them properly when it uses the string, and you'll get unpleasant results.

Figure 6.3
Text/styl resource -32757
from Scrapbook File.

Zen and the Art of Resource Editing

Another place you'll find TEXT and styl resources (you almost never see just one of these without the other) is in AppleScript script applications. If someone else has made a script which is read-only, but you'd like to change the message displayed when the script is started, you can use ResEdit to change it. This is useful if you want to add your own notes about what the script does or how to use it.

Text Resources Overview

The Buddha's Revelations

With the FONT editor, you aren't limited to just characters. You can make letters that look like little pictures, or even symbols from a non-character-based language.

The Macintosh is celebrated for its intuitive graphics: smiling Macs, spinning watches, and trash cans. As graphic as the interface is, we can't escape needing to use text. ResEdit enables you to drastically alter the text that permeates the System, changing it for your amusement or utility.

To change the name of the trash can:

- Open a copy of the Finder with ResEdit.
- Open the STR# picker and double-click STR# 11750.
- Change the first field to the new text.
- Save your changes and replace the original Finder with your new version.
- Restart.

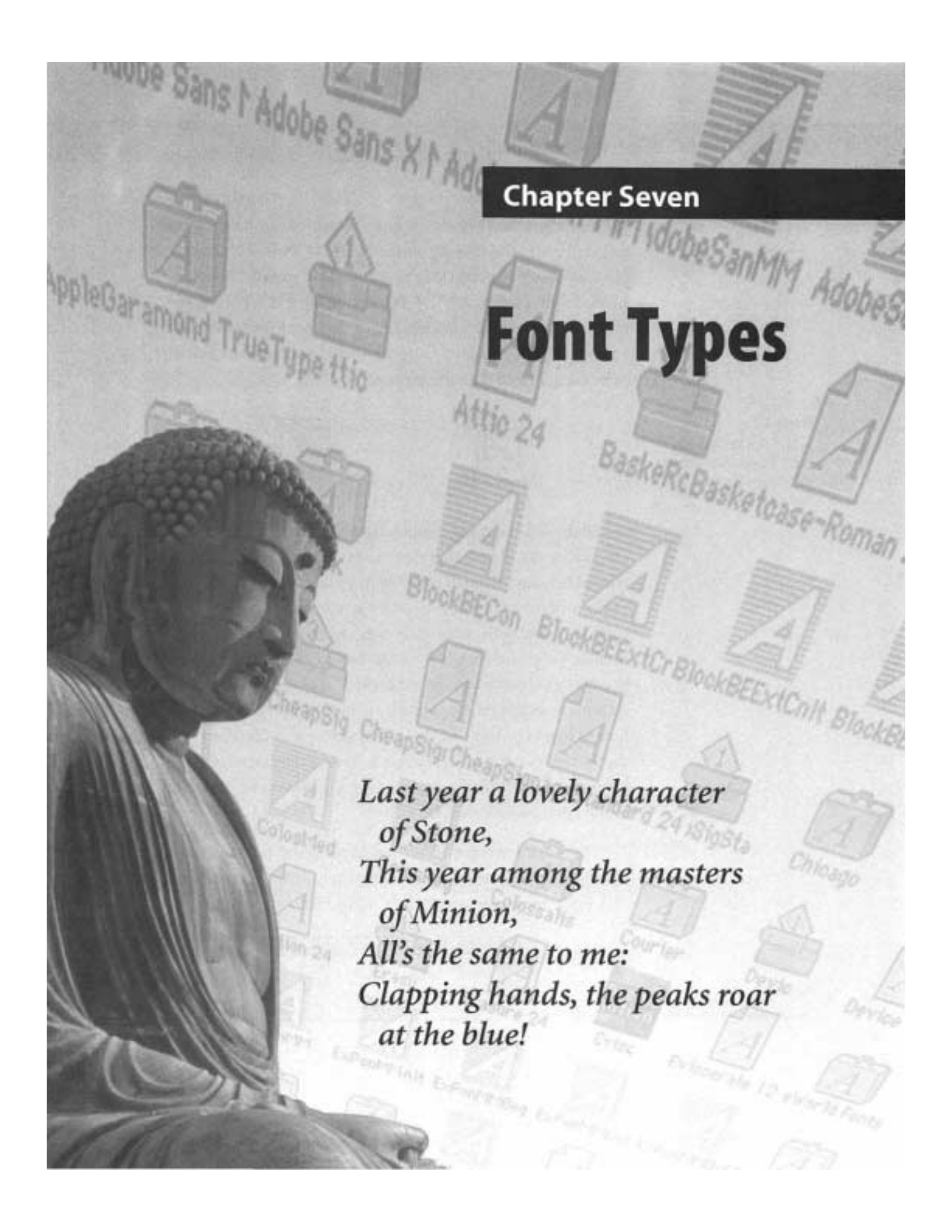
Safety Tips—What Not to Do!

- If you find the word Desktop in your System file, don't change it. This would keep your Mac from being able to start up.
- Don't delete any special characters from strings, because the program won't be able to use them properly and you can get unpredictable results.

The Buddha's Guide to ResEdit Enlightenment

- Can you find a text string that has frogs and gymnasts in it?
- What do you think System STR# -6045 is for?
- What is System 'STR' -16096 used for?
- Can you change Finder STR# 11250 to be something more creative?
- Why do you think the first panel in the default Scrapbook File isn't TEXT- or styl-based?
- Can you translate your entire System to Klingon?

Zen and the Art of Resource Editing



Chapter Seven

Font Types

*Last year a lovely character
of Stone,
This year among the masters
of Minion,
All's the same to me:
Clapping hands, the peaks roar
at the blue!*

The Macintosh font system is a fine example of the phenomenon called thin ice. The friendly surface world of fonts, with its little pictures of suitcases and jolly names such as StoneSerBolita, is rather fragile (see Figure 7.1). Poking too hard at it, putting too much weight on it, or just stepping in the wrong place can send you crashing down into the depths—and the depths are much more complicated than the surface.

Figure 7.1

Geneva 18 is a bitmap font that is usually stored with its entire family in a font suitcase.

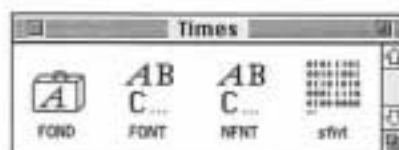


In the olden days of computers, rescue from below the ice was effected by shambling, superhuman programmers who would stare at hex dumps, quaff mightily from tankards of cola, and patch your operating system until it spat you back out onto the surface. Today, in the have-a-nice-day world of the Mac, we have friendly programs with jack-in-the-box icons that give the same powers to nonprogrammers. At the risk of placing a dangerous amount of weight on my thin-ice metaphor, this chapter explains how to use a saw, wet suit, and scuba gear. It also describes some of the murky terrain of the riverbed. It is not a travelogue of the happy surface world; there are many good books and magazine articles that provide that. Let's dive!

Four different types of font resources are on the Macintosh, plus PostScript fonts, which are completely different (see Figure 7.2). Each one is discussed briefly in the following sections.

Figure 7.2

The different font resources.



FONT Resources

Sensibly named, the FONT resource was the original font system introduced with the 128K Macintosh. With this resource, each size of font is stored in a different resource. Thus, Times 18 and Times 14 are two separate FONT resources (see Figure 7.3).

ID	Size	Name
2560	0	Font Family: Times
2574	44	Times 14
2578	44	Times 18

Figure 7.3

FONT resources from the Times family.

In this original font system, every typeface also includes a special FONT resource, which serves as a divider. It has no size, but its name is the name of the font. Thus, when you open a FONT resource, you see several resources that take up no space. This particular resource is used to store the name of the typeface. These resource names are the ones that actually end up being displayed in an application's Font menus.

FOND Resources

Once the LaserWriter came on the scene and users started trying to produce real typography on the desktop, some substantial limits with FONTS became apparent. One of these limitations arose from the fact that there is no provision for typeface families, with true bolds and italics and the like. If you italicize some Times text, all that happens on the screen is that the Macintosh slants the regular Times. If you embolden it, the Macintosh smears the original to make it darker.

If you want to do real typography, these tricks just don't work. In addition to this, the widths of characters in a FONT are specified in points (or pixels), so every character has to be an integral number of pixels wide. However, real characters don't have such even widths, so round-off error is inevitable. This translates into

badly spaced text, because the Mac's and the LaserWriter's ideas of line widths come out different. (You'll still run into this problem if you don't turn on your word processor's Fractional Widths setting.)

The solution was a new resource to contain the extra information needed: the FOND, for FONT Definition. There is one FOND resource for each typeface (see Figure 7.4). Its resource ID is the same as the font ID, and its name is the name of the typeface.

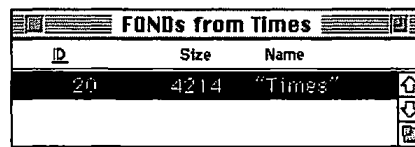


Figure 7.4

The FOND resource picker.

The information in the FOND includes:

- High-resolution equivalents of values previously stored in FONT resources, such as the font ascent, descent, line spacing, and character widths. These are fractional values that can be scaled to any point size. Some of these details we'll cover later in this chapter when we talk about PostScript fonts.
- Flags related to PostScript printing. These mostly tell what to do if some styles, such as a true bold or italic, are missing.
- A Font Association Table (FAT) (see Figure 7.5), which lists the resource IDs of FONTs (and NFNTs) to use for various point sizes and styles (see below).
- A style table, which lists the PostScript font names to use for different styles. This is how the Mac tells the LaserWriter what fonts to use, and how it finds downloadable PostScript font files to send to the LaserWriter.
- Kerning tables for one or more styles.

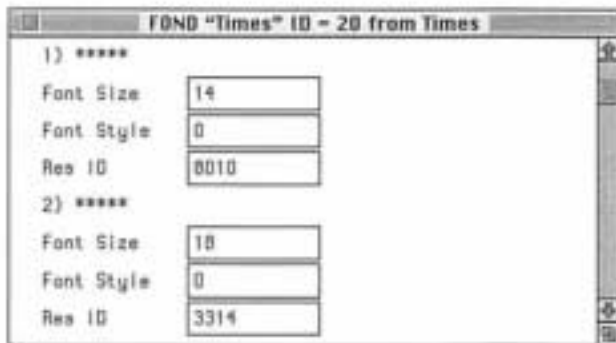


Figure 7.5
The Times Font Allocation
Table (FAT).

NFNT Resources

Even using FONDs, the font system is still limited as long as FONTs are used. Because of their strange numbering scheme, there's still only a tiny range of font IDs: from 0 to 255. This means, of course, that no one can have more than 256 typefaces, including the ubiquitous Chicago and Geneva, installed at a time. If there were more than 256 Macintosh typefaces, some of them would inevitably have the same IDs. This limit was quickly reached, even before PostScript typefaces became widely available.

The obvious solution to this numbering problem is to fix the entire numbering scheme. This requires using a new resource type (to avoid numbering conflicts with FONTs) even though the data inside stays the same. Thus the NFNT, for New FoNT, or New Font Numbering Table, was born (see Figure 7.6). NFNTs contain the same data as FONTs, but because they are referenced only through Font Association Tables (FATs), they can have arbitrary resource IDs. When copying fonts, the Font/DA Mover renumbers individual NFNT resources (updating the FAT references) to make sure that no two in the same file have the same ID. This renumbering causes no harm to documents, because NFNT IDs are normally used only by the internals of the Font Manager.

ID	Size	Name
5674	5534	"Times"

Figure 7.6
The NFNT resource picker.

Freed of the 256-font limit, typefaces can have font IDs up to 16,383—as long as the bitmap fonts are contained in NFNT resources. (The range from 16,384 to 32,767 is reserved by the Script Manager for fonts of other writing systems such as Hebrew or Kanji.) Apple registered font IDs, making sure that no two fonts, even from different vendors, have the same ID. Everything seemed fine until mid-1990, when it came to pass that the entire range was used up. Now we're back in the same boat: There are different fonts with the same IDs, and as a result different people's systems may have different IDs for the same fonts.

sfnt Resources

The sfnt is the latest addition to the family. sfnts are outline fonts, like PostScript fonts. This means that they contain mathematical descriptions (known as spline curves) of the outlines of characters. Apple's TrueType technology (built into System 7 and available as an INIT for System 6.0.7) can scale these outlines up to any point size and fill them in with pixels, resulting in smooth, well-shaped characters at large point sizes (see Figure 7.7). This is a capability that bitmap fonts, such as those controlled by the FONT resource, sorely lack, as anyone knows who's tried to use a size that is not available.

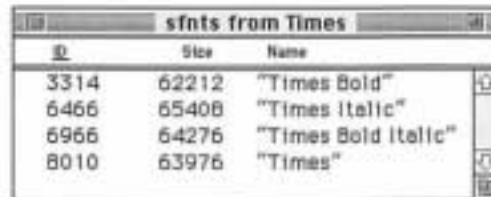
Figure 7.7

This is a TrueType font, which can also usually be found in a font suitcase with its family.



TrueType and sfnt resources provide the same immediate advantages as Adobe Type Manager. However, sfnts are better integrated into the operating system, making them easier to install

and more amenable to future growth. No separate downloadable PostScript font is required; the sfnt resource lives in a suitcase file or the System file, just as other font resources do (see Figure 7.8).



ID	Size	Name
3314	62212	"Times Bold"
6466	65408	"Times Italic"
6966	64276	"Times Bold Italic"
8010	63976	"Times"

Figure 7.8
The sfnt resource picker.

What's in a Font

A font or typeface (the two terms have become almost interchangeable) is a set of images of characters. Because the characters are intended to be placed next to one another in a line, the resource includes not only the shape of each character, but also its vertical alignment relative to the baseline and horizontal spacing relative to adjacent characters. (The alignment and spacing are called *metrics*—see Figure 7.9.)



Figure 7.9
Alignment and spacing font metrics.

There is also information that applies globally to all the characters, such as the leading—the distance between lines of type in a font. Technically, leading is the amount of extra space between lines plus the point size, measured from the baseline of one line to the baseline of the next line. For example, 12 point type with a 14 point space between lines (often called 12/14) has 2 points of leading.

Also, a font may have a table of kern pairs, which are adjustments to the spacing between specific pairs of characters, such as “A” and “V.”

And, of course, every font or typeface has a name.

Beyond these attributes, there is also an internal set of linkages that connect fonts to one another. In the old days of just the FONT resource, there had to be a linkage between different sizes and styles of the same typeface. When you change text to 18 point Times Bold, the Font Manager within the Macintosh can follow these linkages to find the appropriate set of bitmaps, or the appropriate TrueType outline, to display.

Editing Font Resources

ResEdit 2.1 provides a graphical editor for FONT and NFNT resources that enables you to modify or create bitmap fonts. It provides a textual, template-based editor for FONDS that enables you to edit some parts of the FOND but not others. It has no editor for sfnt resources, which are extremely complex in structure.

To see what the FONT editor looks like, open a copy of your System, and open FONT resource 396 (Geneva 12). If you have a color machine, you’ll get a dialog informing you that changes you make to the FONT will not be displayed in the sample text in the editor. This is an unfortunate bug in ResEdit, which the programmers have handled by politely informing you of its presence.

The FONT/NFNT Editor

The FONT editor has several important components (see Figure 7.10). As with all bitmap editors, the largest part of the editor is a FatBits view of the character you are editing (in this

case, the letter “A”). To choose a different character, simply type it on your keyboard. The two upward-pointing triangles below the character mark the left and right sidebearings of the character—its left and right margins. Dragging these outward increases the amount of space around the character; dragging them inward reduces it. It is perfectly all right for the character to stick out past the sidebearings on either side; this means that it overhangs neighboring characters. Many italic letters do this.

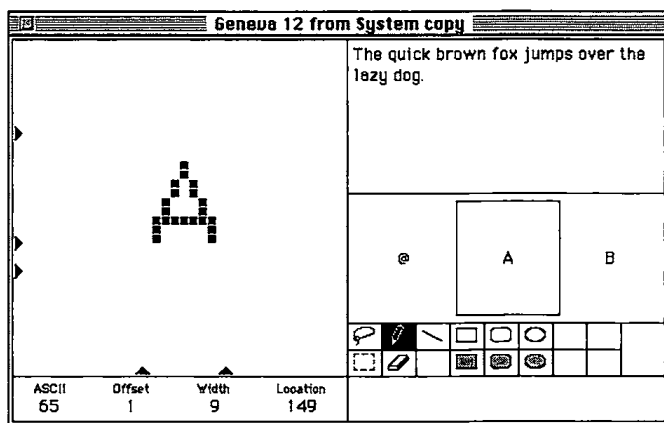


Figure 7.10
The FONT editor.

The baseline marker is the middle of the three triangles to the left of the character. It marks the baseline, the line on which all characters in a line, regardless of font, size, or style, sit. It cannot be moved.

The ascender and descender lines are the other two triangles on the left side of the editing area (the ascender is the top triangle); these mark the top and bottom of the entire font. No part of the font can go above the ascender or below the descender. (If you move these markers too far in, you will end up chopping off the tops or bottoms of letters. Be careful!) The ascender and descender are occasionally used to determine the line spacing of the font, but modern applications get this information from the FOND or make it up themselves.

Below the editing area are some numeric displays. These show more-or-less technical information that is probably not of much use to the more casual user. The ASCII code is the internal code (0–255) of the character (for more information on ASCII codes, check out Chapter 8, “Tuning Keyboards”). The offset is the distance from the leftmost black pixel to the leftmost left-sidebearing marker of the entire font (sorry you asked?); the width is the set width of the character (the amount the “pen” moves when drawing that character, equal to the distance between the left and right sidebearing markers); and the location is the offset to the character’s bitmap in the font’s internal storage.

Last, there is a sample text display at the top right, so you can see how the characters look in context. If your screen is in black-and-white mode, any changes you make to the characters are reflected in the sample text. You can edit the sample text by clicking it; then you can type or you can drag to select text, as in any other text field. Clicking outside the text box removes the selection or insertion point. From then on, typing switches to the character you typed for editing.

The FOND Editor

The FOND editor is textual and based on a template. It’s of limited use: It doesn’t let you edit the kerning or style-mapping tables, and if the resource is too large, it refuses to open it at all. Still, it’s very useful to be able to edit the font ID and the Font Association Table (FAT). In working with the FOND editor, keep in mind that many numeric values are shown in hexadecimal (base 16). These are the numbers that start with a “\$” and may include as digits “A” through “F.” You can enter new decimal values in these spaces; just make sure to delete the “\$.” Similarly, if you enter a new hexadecimal value, make sure you keep the “\$” before the number.

Most of the fields should remain undisturbed, but some can be very useful. For instance, several fields enable you to set extra widths for different styles (bold, italic, and so on). The FOND resource also has fields that can be used to set global values for the ascender and descender of a typeface, no matter what the size. Near the end of the FOND resource (directly before the field labeled The Tables), you can find the Font Association Table—the FAT.

The FAT lists the resource IDs of screen fonts (FONTs, NFNTs, and even sfnts) to use for various point sizes and styles within a FOND (see Figure 7.11). Thus, when the computer needs to display text in a particular font, size, and style (for instance, 12 point Times Italic), it looks to the FAT area of the appropriate FOND to find which FONT, NFNT, or sfnt to use for that size and style. By including style information, it makes real screen fonts for italic and bold styles possible.

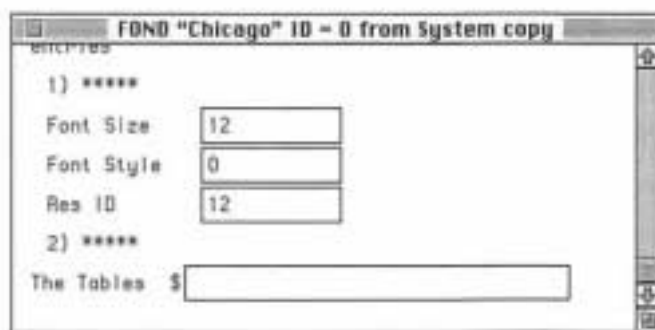


Figure 7.11

The FAT (Font Association Table) for the System font Chicago.

The FAT is a typical ResEdit list. Each entry in the list (which often contains a single entry) consists of a point size, a style code, and the resulting resource ID. The point size is the actual size of the font to which you are referring. The style code is the result of adding together various style values:

Zen and the Art of Resource Editing

0	Plain
1	Bold
2	<i>Italic</i>
4	<u>Underline</u>
8	Outline
16	Shadow
32	Condensed
64	Extended
256	4-color font
512	16-color font
768	256-color font

Following this formula, Bold Condensed style would be $1 \text{ (Bold)} + 32 \text{ (Condensed)} = 33$. In practice, only the Bold and Italic bits, and sometimes Condensed and Extended, are used.

NFNTs are accessed through the FAT and can be given any resource IDs, so long as those IDs don't conflict with any other NFNTs. A FONT resource can't be given a different ID just because it's being referenced through a FAT. It must still use the $128 \times \text{font ID} + \text{size}$ formula. (For this reason, a typeface that uses any FONT resources cannot have a font ID greater than 255.)

If you renumber FONT or NFNT resources associated with a font, you must update the entries in the FAT.

It's critically important, if there's any data at all shown in the box labeled The Tables that follows the FAT, that you don't insert or delete entries from the FAT list. Changing the size of the FAT invalidates internal pointers in the FOND tables, which then destroys its data and causes horrific effects, including system crashes, any time the System tries to use that font.

It's also very important that the entries in the FAT remain sorted. The point sizes are listed in increasing order, and within a point size, the style codes are listed in increasing order. If you don't do this, you seriously annoy the Font Manager, and it returns many kinds of wonderful System errors.

You can use the FAT to change your System font. If you open the FOND resource for Chicago in the System file, you'll notice one field that lists the size as 12. This is Chicago 12, and this is what the System looks for when it wants to know what FONT to use. Normally, the corresponding ID number is 12, which is the Chicago 12 installed into the System and the ROMs. However, there's no reason why the ID number can't be the ID number of another FONT. Simply type the new ID, having found it from the FONT resource, and keep the size as 12 (though your ID number can represent a font of any size, because each FONT represents a certain size). There are some problems with doing this, however. One comes from making the new font too big. The Macintosh is nice enough to resize your menu bar and menus as needed to accommodate the font that is used for the System font, but it doesn't resize the dialog boxes. As a result, you'll get a dialog box that is the same size, with all the same information, but in a much larger font.

Editing sfnts...Not!

ResEdit doesn't provide an sfnt editor, and probably won't in the future. sfnts are extremely complicated beasts, and editing spline data requires very sophisticated graphical editors. The only practical way to work with sfnts is to buy a dedicated type design program such as Altsys' Fontographer, Letraset's FontStudio, or Kingsley/ATF's ATF Type Designer. Techies who simply must know how the data is organized should contact APDA (the Apple Programmer and Developer Association) at (800) 282-2732 and

The Buddha's Revelations

If you're using QuickDraw GX, you probably already know that PostScript fonts have to be enabled in order for you to use them. What the System does is copy the information from the POST resource and translate it into an sfnt resource. Presto! Your PostScript fonts are now indistinguishable from TrueType fonts.

Zen and the Art of Resource Editing

You won't find any sfnt resources directly in your System file. TrueType fonts are stored in their own files, usually in the Fonts folder.

order the TrueType Font Format Specification, part number M0825LL/A.

There is actually one useful thing you can do with an sfnt resource: You can see the author and copyright information for the font. If you open the resource, you get ResEdit's generic hexadecimal editor. By scrolling through the data, you can find the author and copyright information. It's hard to miss; it's about the only human-readable data in the entire resource. Look in the text column on the right side of the editor. The text is usually located near the beginning or the end of the data. This is handy if you've obtained an ostensibly public-domain or shareware TrueType font and you want to make sure it's not really a commercial font. Some people have occasionally converted commercial PostScript fonts to TrueType using utilities such as Metamorphosis or FontMonger. The resulting TrueType fonts are still commercial, and distributing them is software piracy. If you look at a "shareware" sfnt and find a copyright notice that says, for example, "Copyright ©1988 Adobe Systems, Inc.," that's what's happened. Get rid of it and inform the person you got it from.

Renaming or Renumbering a FOND

In all modern Macintoshes, it is the FOND resource that gives a typeface its name and font ID.

If you want to rename the typeface, all you have to do is use the Get Resource Info command to change the resource name of the corresponding FOND resource. Keep in mind that all applications that store font names in documents (such as PageMaker) will no longer recognize the font name in existing documents, and will probably change the type to Geneva or Courier.

To renumber a typeface, change the resource number of the FOND. Then open the FOND and change the Family ID field to

the new number. (If all the bitmaps for the typeface are stored in NFNTs, Apple suggests you use a new font ID in the range 256–1023, because that range is reserved for renumbered typefaces.)

If some of the bitmaps for this typeface are stored in FONT resources, you'll have to renumber all the FONTS because their numbering is based on the font ID. For each FONT, compute the new resource number based on the new font ID. The formula for FONT IDs is $(128 \times \text{Font-ID}) + \text{Point Size}$. Renumber the FONT resource accordingly and change the resource number stored in the FAT.

Renumbering incurs the same problem as renaming: Applications that rely on font IDs will lose references to the font stored in existing documents.

For people who use Microsoft Word: Never renumber the Symbol font if you plan on using Word's formula commands. Word assumes that the Symbol font's ID is 23 when it displays math characters in formulas. If this is no longer true, you will get garbage characters, most likely in Geneva, in your formulas.

Inside PostScript Font Files

Downloadable PostScript font files aren't usually very interesting or informative to look at, but we'll describe them here for the benefit of those of you who simply must know everything. These files are just containers for the PostScript code that makes up a downloadable font (see Figure 7.12). When a document being printed to a PostScript printer makes use of a PostScript font that isn't already living inside the printer, the Mac looks (by name) for a downloadable file for that font. Then, it sends the enclosed PostScript code to the printer. The result is that the font is now living inside the printer and can be printed.

Figure 7.12

A typical PostScript font file. These require an accompanying bitmap font for the System to be able to recognize.



Those Weird Filenames

The filenames of downloadable PostScript fonts are a great source of amusement to nearly everyone. They're also very important, as you will agree if you've ever renamed one and then tried to print with that font.

When the Mac looks for a downloadable file for a particular font, it looks for it by the name of the PostScript font. Unfortunately, Macintosh filenames can't be longer than 31 characters, and some PostScript font names can exceed that limit. So the name of the downloadable font file is compacted. It uses the first five letters of the first word of the font name and the first three letters of each following word (this is called the 5-3-3 rule). Thus, the PostScript font called Stone Serif Bold Italic resides in a downloadable file called StoneSerBolIta.

What's Inside

Downloadable files contain only a few resources (see Figure 7.13). Most, such as the BNDL, FREE, and ICN#, are just there to give the file its icon in the Finder. All the PostScript code is contained in POST resources.

The POST resources start with number 501 and increase from there. Each resource contains a flag byte, another byte of filler, and then the data. (The template provided by ResEdit is wrong. When you open POST resources, use the Open Using Hex Editor command to open them. Then you'll just get a hex dump.)



Figure 7.13
The resources within a
PostScript font file.

If the flag byte is 1, the rest of the data is plain text. POST 501 is almost always of this variety. The following text is the PostScript font header, which contains some interesting stuff such as the PostScript font name, the font type (1 or 3), and the copyright notice.

Most of the succeeding resources have a flag byte of 2, which indicates compressed binary data. This is basically unintelligible to mere mortals.

The last resource has a flag of 5, which indicates the end of the data.

Some fonts have a POST resource with a flag of 3. This means that the rest of the PostScript data is in the data fork of the file and isn't stored in resources at all.

Uses for Bitmap Fonts

There are still a few uses left for new bitmap-only typefaces. One use for the font resources is to add a character to a font. If you have a certain character that you would like to use and it isn't in a font, you can just add it to the appropriate FONT resource (if the font is held in a FONT resource, such as the Macintosh System fonts). Keep in mind that because each FONT represents one size, the special character will be available only in that size. To use it in all the sizes, you would have to add it to each separate FONT resource.

Another possible use for font resources is to add a new size to an existing font. This takes some time, but it can be worth it if a font you really like does not come in a desired size.

Font Overview

You can see that the Macintosh font system has gotten more complicated over the years. What started as a simple little resource has become a set of resources dependent on each other for survival. While editing fonts with ResEdit has become less practical, knowing how they relate to your system and each other is a valuable tool for understanding the way the Macintosh itself works.

To get to the font resources:

- Open ResEdit and use it to open a program you want to look at.
- Double-click the FONT, FOND, NFNT, or sfnt resource.

To edit a FONT or NFNT:

- Double-click the specific resource you want to edit (for example, FONT 12).
- Type the character you want to edit to bring it into the display area.
- Use the Paint tools to edit the character.

To add a FONT to a FOND resource:

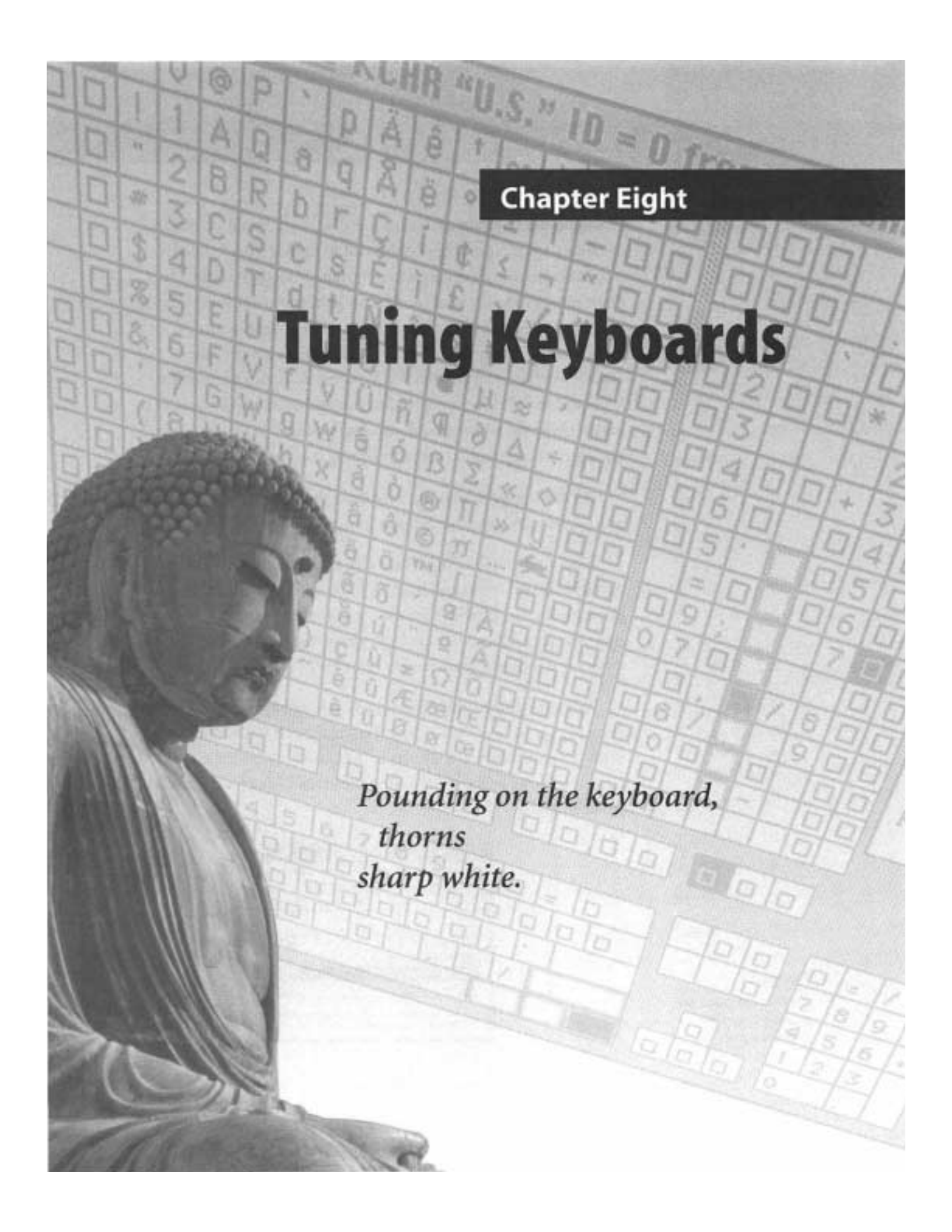
- Double-click the FOND resource you want to add to.
- Scroll to the area where your FONT will be added (remember that all fonts must be added in increasing order).
- Click the row of asterisks prior to the next-highest size of FONT.

- Choose Insert New Field from the Resource menu.
- Enter the required information.

The Buddha's Guide to ResEdit Enlightenment

- When are POST resources made into sfnt resources?
- Why are TrueType fonts quadratic-based?
- Why are PostScript fonts Cubic-based?
- How many characters are there in a Macintosh font?
- When is a font a fountain?
- On Power Macs, what are rdhr font resources?
- How do multiple master Type 1 PostScript fonts work?
- What are PostScript Type 2 and Type 4 fonts?
- What happens when you print a TrueType font to a PostScript printer?
- Can you find a little lamb, a jackrabbit, and a black bird in Geneva?

Zen and the Art of Resource Editing



Chapter Eight

Tuning Keyboards

*Pounding on the keyboard,
thorns
sharp white.*

Changing the function of the keyboard is an extremely common thing to want to do. Some people want to keep the period and comma keys from changing when they hold down the Shift key. Or because of the unusual configuration of some Apple keyboards, sometimes people are curious about switching the Control, Option, and Shift keys. Or because of the inefficiency of the Qwerty keyboard layout, they might want to change their entire keyboard to the Dvorak configuration (and then usually change it back). Many early PowerBook owners want to disable their Caps Lock key to keep from constantly hitting it accidentally.

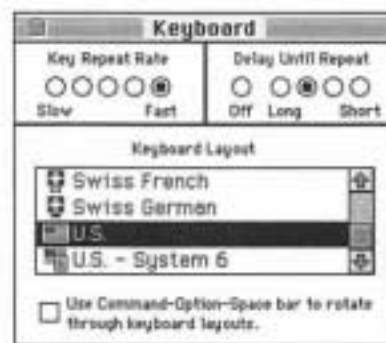
All of these problems can be solved by using ResEdit to create or modify a keyboard mapping resource called the KCHR.

Under System 7.5, Apple provides an extensive set of international keyboard layouts all built with the KCHR resource. There is also a layout that provides System 6's keyboard map for System 7 users who need it (we haven't been able to figure out what it does differently, though, so we tend to ignore it).

When you have more than one KCHR installed in your System, you can use the Keyboard control panel to choose among these different keyboard layouts (see Figure 8.1). Note the small icon next to the keyboard layout in the list of layouts; we'll talk about this later.

Figure 8.1

The Keyboard control panel, showing some of the keyboard layouts installed.



Behind the Scenes

Before we get into editing a KCHR, you need to have an idea of the hoops the Mac jumps through every time you touch a key on your keyboard. These hoops, in brief, are: You type a key, the Mac gets a raw hard-wired number called a “key code”; this code is translated via a “translation table” into a character. There are multiple translation tables and which one the Mac uses depends on the combination of modifier keys (such as Shift or Control) you are holding down—if any—when you press that key. ResEdit enables you to modify these translation tables, thereby changing the character that appears when you type a specific key with or without a specific modifier key. Let’s look at this system again in more detail.

Each key on your keyboard, whether it’s a character key or a modifier key such as Shift, has a numerical “raw key code.” The system uses a KMAP resource to translate the raw key code into a “virtual key code.” For most keys, the virtual key code is the same as the raw key code. Generally, when we are talking about a “key code” through the rest of this chapter, we are referring to the virtual key code produced by the KMAP. ResEdit does not have an editor for the KMAP resource—you don’t need to worry about it anyway.

After a keyboard action has been translated to a key code, it is translated to a character via the KCHR resource your System is currently using. Within a KCHR resource are several tables, each containing all the keys on the keyboard. Any combination of modifier keys (including none at all) is associated with a particular table. When you hold down a particular modifier key, its table is activated. Because there are five modifier keys (Shift, Option, Command, Control, and Caps Lock), there are 32 possible combinations of modifiers, so there could be up to 32 tables in a

For some international Systems and keyboards, the key codes are further rearranged using an itlk resource, but the U.S. System does not use an itlk.

KCHR. However, the standard U.S. KCHR has only eight translation tables. To avoid the extensive repetition involved in having a table for every possible combination, the KCHR tables can be linked to more than one modifier or modifier combination.

For instance, translation Table 0 is used for no modifier, for Command, and for Command-Shift. While pressing the Command key usually causes your software to do something different when you press a letter key, the Command key doesn't rearrange the characters on the keys, and so it produces the same character as not holding down any modifier key at all. Similarly, holding down the Caps Lock key corresponds to a different translation table than holding down the Shift key. This is because the Caps Lock key has capital letters and normal numbers, but the Shift key corresponds to a keyboard layout that consists of capital letters and the symbols above the number keys.

The KCHR Editor

When you open a KCHR resource in its editor, you get a large window filled with little boxes and a diagram of a keyboard (see Figure 8.2). This diagram dynamically hilites when you hold down keys on the keyboard to show you the character, its key code, and the translation table used.

*Hilite is the new computer-world version of **highlight**. Or it is simply the Kleenex of day-glow pens. Either way, we choose to use it in our goal to cultify ResEdit.*

When this screen shot was taken, the Command and Option keys were held down, causing those keys to be hilited in the keyboard layout diagram at the bottom of the window, and causing the appropriate translation table, Table 6, to be selected and displayed at the upper right.

In the upper left of the picture, there is a table of the ASCII character set, laid out in a 16×16 grid. This grid cannot be changed—after all, this isn't an ASCII editor—except for the font used in the display, which is handy for finding out if the font has

any interesting characters. When you press a key on your physical keyboard, the character to which it is mapped in the virtual keyboard hilites in this grid. You can also click and drag the characters from this grid to one of the other grids, which we will get into when we start editing.

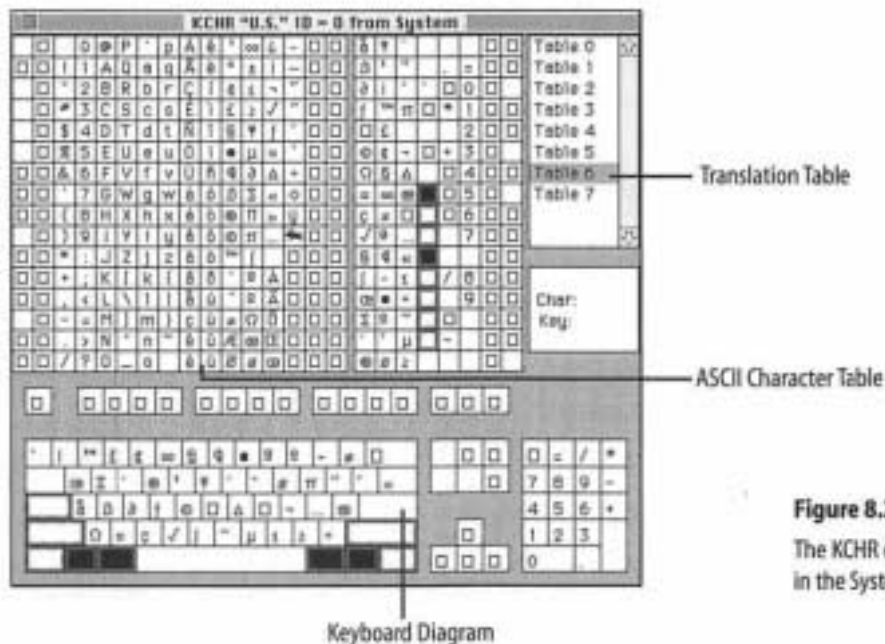


Figure 8.2

The KCHR editor for KCHR 0 in the System.

To the immediate right of the character grid is an 8×16 grid. The positions in this array represent key codes. The character at a given position is the one used with a particular key code and with the active translation table. For instance, in Figure 8.2, Table 6 has an infinity sign in the eighth row and second column. This means that the key code \$17 (that's hexadecimal, folks, and you'll only see it when you actually click the square in the translation table) gets translated to the infinity character when the Command and Option keys are held down.

Remapping Ordinary Keys

Let's say we want to make Shift-period send a period instead of a greater-than sign. First, you need to open the KCHR you want

ASCII is the international method for assigning numeric values to the roman character set. It contains 128 characters and is generally compatible with everything. The Mac uses an extended ASCII set called Unicode that contains 256 possible characters, allowing more accented vowels and other symbols.

Zen and the Art of Resource Editing

to edit—we suggest you use KCHR resource 0 from a copy of the System file—the “U.S.” layout.

Then you must locate the period in the character array by pressing the period key to hilite it. Let go of the period key and press and hold the Shift key. Click the mouse down on the period in the character array and drag it to the place you want it on the keyboard diagram at the bottom of the editor (that is, right on top of the greater-than symbol) (see Figure 8.3).

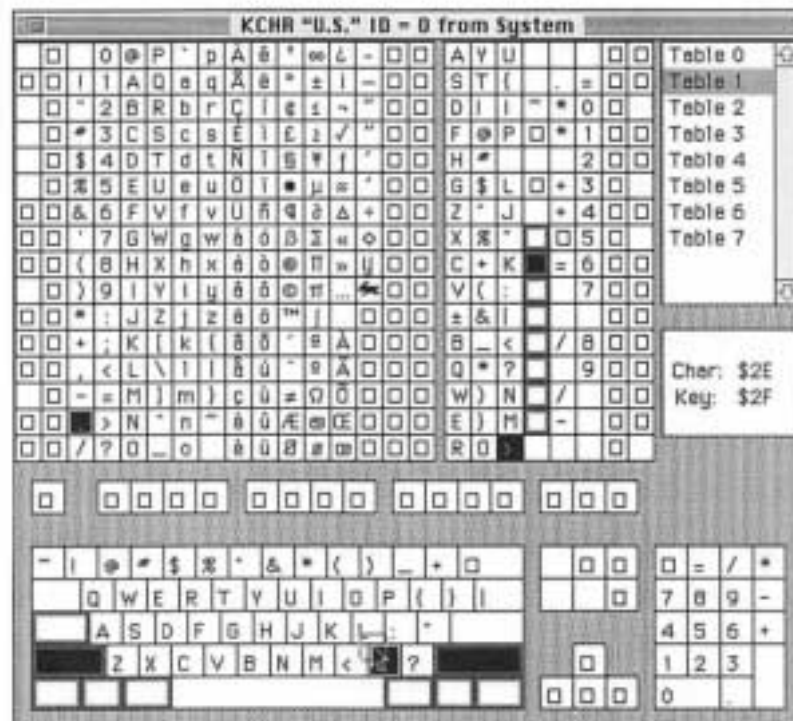


Figure 8.3
Dragging the period
character to the keyboard
diagram.

Release the mouse and then release the Shift key. Save your changes and go through the now-routine task of switching your edited System file with the active System and restart. Now when you type a Shift-period, you will get a period!

Remapping Modifier Keys

Recalling that the various tables shown in the KCHR window correspond to certain combinations of modifier keys, you can easily experiment to discover the correlation shown in the following table.

Table	Combination of Modifier Keys
0	No modifier, or Command and Command-Shift (also with Caps Lock)
1	Shift
2	Caps Lock
3	Option
4	Shift-Option
5	Caps Lock-Option
6	Command-Option
7	Control (with any other modifiers)

Let's suppose we want to exchange the roles of the Shift and Control keys. This requires a number of table reassignments. Working on yet another copy of the System file and on KCHR resource 0, start by reassigning the Control key to translation Table 1. To assign a modifier key to a translation table, hold down the key and click the table name in the table list. ResEdit then shows an alert asking if you really want to do this, as shown in Figure 8.4.

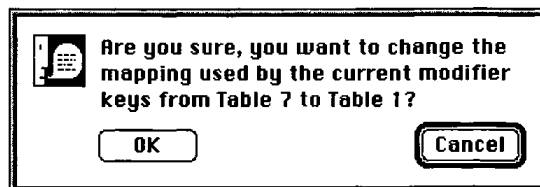


Figure 8.4

ResEdit provides a warning when you remap a modifier key to a new translation table.

After you replace Table 1, you also need to reassign Shift to Table 7, Control-Option to Table 4, Command-Control to Table 0, and so on. If you can't figure out the rest, we suggest you don't attempt this at all.

As you can see, remapping the modifier keys can quickly lead to a little ResEdit nightmare, so don't start planning anything wild until you've got it completely figured out. For example, let's look at how you might disable the Caps Lock key, a useful edit if you've got a PowerBook.

Unlike the first example of switching two of the modifiers keys with each other, this project involves either trying to get rid of all references to one of the modifiers, or trying to find a way to hide it by adding it to all the other layouts so it doesn't cause a table change when activated. While it should be obvious that the former path is as involved as removing small bones from a fresh-water trout, we'll give up and just eat the whole thing. This also gives us a chance to explore all the wonderful menus of the KCHR editor!

The KCHR Editor Menus

Getting rid of the Caps Lock tables should be easy enough. For instance, Table 2 is just the Caps Lock key by itself. Hold down the Caps Lock key (yes, it does this by itself) and click Table 0 to move the Caps Lock key to this no-modifiers table. ResEdit asks you if you really want to do this; click OK. After you've done this, you can use the Remove Unused Tables command in the KCHR menu to remove Table 2. This causes all the following tables (3 through 7) to each move up one position just to completely confuse things. So Tables 3, 4, 5, 6, and 7 are now Tables 2, 3, 4, 5, and 6 respectively.

The only other table that utilizes the Caps Lock key is Table "5" (now Table 4), Caps Lock-Option. To remove the effect of the

Caps Lock key on the Option key, reassign the Caps Lock-Option combination to the Option-only table—Table “3” (now Table 2!).

After you’ve done this, you can test all the key combinations and check the tables they hilite to see if the Caps Lock key has any effect on them. Now you can choose Remove Unused Tables again to delete Table “3” (still actually 2).

Here is the first of the little bones you have to just swallow—Table “3” didn’t go away! Why not? You can’t find any key or combination of keys that activates it, so it must be unused. What’s wrong? Well, it has something to do with the fact that the KMAP knows that there are actually two Option keys (on most Apple keyboards) and it can’t figure out by itself that you want to add the Caps Lock key to both of them, so it gets stuck. The only way to get rid of this table is to use the Uncouple Modifier Keys command and edit the KMAP—both of which are beyond the scope of this book. To quote *Inside Macintosh*, “This capability is included for compatibility with certain existing operating systems that distinguish the left and the right keys. Its use by new applications violates Apple user interface guidelines and is strongly discouraged.”

Our Zen-like solution is to not worry about the extra table that doesn’t really get used, although something in there somewhere is hanging on to it with a tight grip. We’ll just go about our merry way.

At this point, we’ve sort of used the Remove Unused Tables command, and we’ve talked about not using the Uncouple Modifier Keys command. What are all those other KCHR menu commands for, anyway?

The first KCHR menu item, View As, is handy for viewing keyboard layouts of different Apple keyboards. You’ll find a whole lot of layouts, from the original 128K Macintosh all the way

through some of the latest PowerBooks. Each of these layouts is defined in a KCAP resource which is also not easily modifiable.

Skipping down, you'll next find the New Table, Duplicate Table, Remove Unused Tables, and Remove Duplicate Tables commands. These are all pretty straightforward commands for manipulating the tables in any KCHR. You'll no doubt notice the absence of a Remove Table command, which would make the whole bit earlier about reassigning a key combination to another table all moot.

Dead Keys

No, no. Not the type you might find on old keyboards! Dead keys in the KCHR resource are keys that don't do anything until a second key is pressed. This is most commonly used for all those non-English-like accent marks, which require you to do something like press Option-E and then E in order to get é. Dead keys enable the Mac to give you all those cool accent marks without needing a separate key for each one.

In the U.S. keyboard layout, there are five of these "dead" keystrokes. They are Option-E (known in English as an acute accent), Option-I (circumflex), Option-U (dieresis), Option-N (tilde) and Option-*grave* (`). Only certain characters can take these special accents. Typing an option-E followed by a W results in ´w, not the obvious alternative. If you want to use the accent mark by itself, you can type the dead key twice and it produces just the diacritical.

If you type a dead key while in ResEdit's KCHR editor, it brings up a special dead key editing window. Within the dead key editor is a large character array just like the one in the main KCHR editor (see Figure 8.5). To its right there is a list of character pairs. Each pair is simply a character without and then with its accent. In

the upper right of the editor window is a lone character that is the accent that is created by the particular dead key you are viewing.

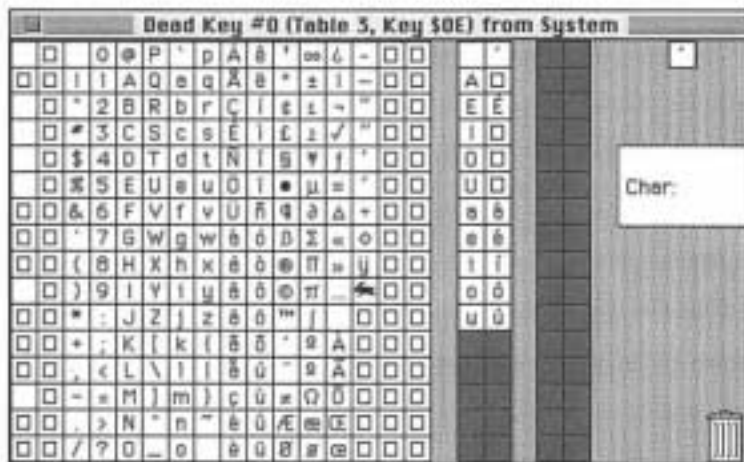


Figure 8.5

The dead key editor that is brought up when you type Option-E in the KCHR editor.

Let's add a bogus accent character to see how to edit a dead key. Let's say you want to define an accented x to be the infinity symbol. Drag an x from the character array to the left-hand side of one of the unoccupied pairs, and then drag an infinity character to the right-hand spot. Now when you press Option-E followed by X, you'll get the infinity symbol. (To delete or get rid of any of these pairs, you can just grab it and drag it to the trash can icon in the lower right.)

You can use the Edit Dead Key command in the KCHR menu to bring up a list of all the defined dead keys, from which you can select one to edit (see Figure 8.6).

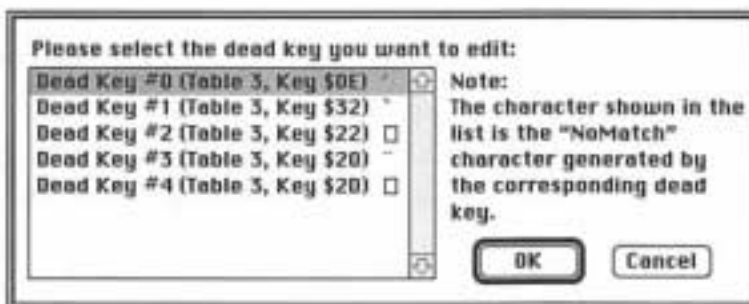


Figure 8.6

ResEdit's Edit Dead Key command from the KCHR menu.

You can kill a key by holding it down and choosing Convert to Dead Key from the KCHR menu. This adds it to the list and you can then add pairs of characters to combine. Some fonts use dead keys for other types of accents, such as a chess font that adds an x across the character/piece to indicate that it's captured.

The last command in the KCHR menu is Remove Dead Key. This command is available when you are viewing a dead key's editor. It quickly revives the dead key to an active key with its default accent.

KCHRs in International Systems

Keyboard layouts are controlled by a part of the Macintosh System called the Script Manager. The Script Manager is responsible for the Mac's ability to work with different languages and writing systems. Each writing system, or "script," has an identification number. For instance, the roman script used for most European languages is number 0, Japanese is number 1, Arabic is number 4, and Vietnamese is number 30.

Each script has a range of possible KCHR ID numbers. The roman script gets the lion's share of potential KCHR ID numbers, 0 through 16383. Each other script has a range of 512 KCHR numbers, starting from $16384 + 512 * (\text{the script ID number} - 1)$.

For each script, there is a currently active KCHR. The KCHR for the next time the System starts up is stored in an itlb resource whose resource ID is the script's number.

The roman script is always available. When another script is also available, the small icon representing the active keyboard layout appears in the menu bar. You can switch scripts by clicking the icon or by typing Command-Spacebar. There are also command keys that simply set the script to roman or set it to the other

If this key doesn't work to change layouts, you may be running System 7.5, which added a checkbox option in the Keyboard control panel to turn this on and off.

script. (These keystrokes are specified in the KSWP resource.) To change keyboard layouts within a script, use Command-Option-Spacebar.

Small Icons with KCHRs

Earlier, we discussed the fact that a KCHR could have a small icon (or SICN) associated with it. This is useful if you want visual feedback about what keyboard layout you're using.

If you are running System 6, you can display the small icon of the current KCHR in the menu bar by modifying the itlc resource in the System file (see Figure 8.7). Just set the Always Show Icon radio button to 1 or true. This applies to System 7 as well; however, System 7 uses a new type of icon especially for keyboard layouts—the kcs# and kcs4 resources.

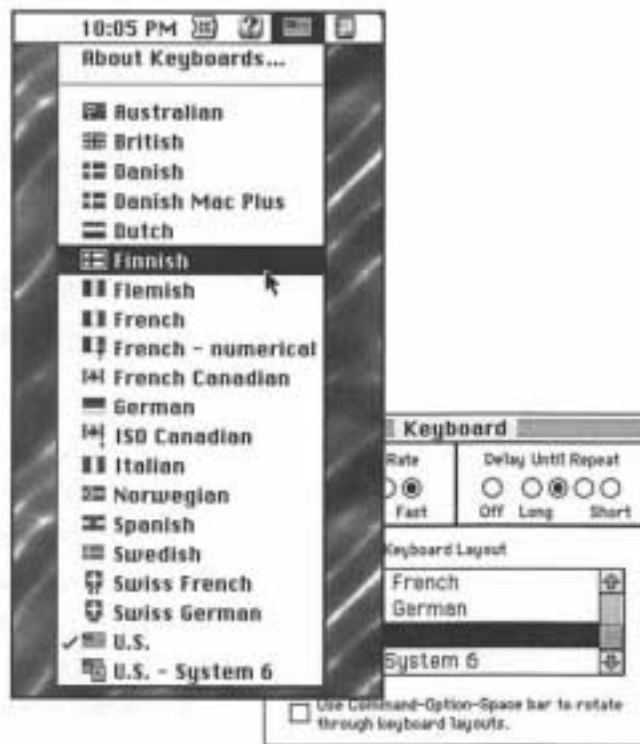


Figure 8.7

The Keyboard menu showing all the installed keyboard layouts.

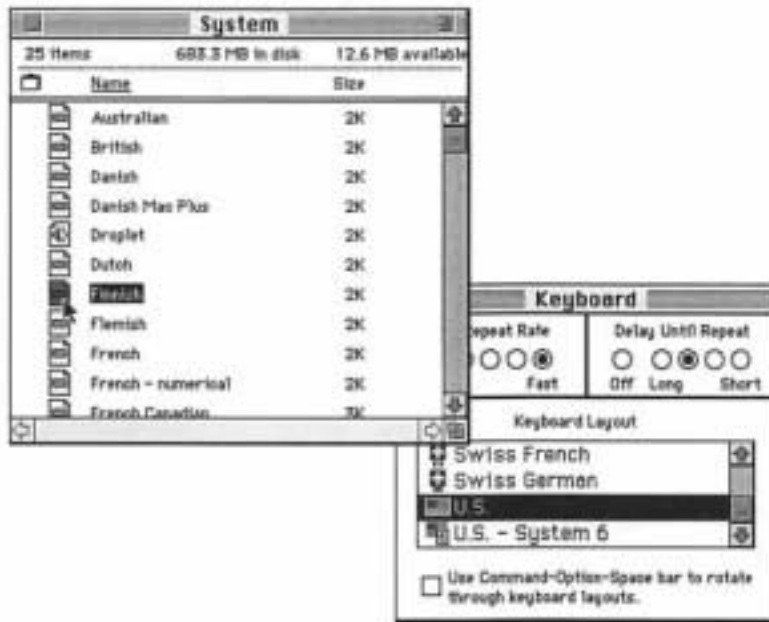
Unfortunately, ResEdit by itself doesn't have an editor for this resource. There is an editor created by Apple for developers, but because it is unsupported, it cannot be licensed for distribution with books or other products. You may be able to find it on various online services or from an Apple FTP site. Check out Chapter 15 about ResEdit and its resources for information on installing and using custom editors.

If you're running System 6, you have an advantage (sort of). System 6 only uses a SICN as the KCHR icon, which you can easily modify based on your experience in Chapter 3. To associate the SICN with a particular KCHR, you only need to make the resource IDs the same.

Keyboard Layouts in the Finder

With System 7 came the ability to double-click the System file "suitcase" to open it and peek inside. You couldn't do everything with this ability, but you could use it to easily install and remove fonts, sounds, and keyboard layouts. Of course, in System 7.1 and later, you deal with fonts via the Fonts folder, but this is still the easiest way to install sounds and keyboard layouts. If you want to remove a keyboard layout (perhaps if you're not likely to use the Flemish keyboard), you can open the System suitcase and drag the keyboard layout to the trash. No need for ResEdit or any other utilities. It's just built-in.

Any KCHR resource you create in the System is visible as a keyboard layout item in the System suitcase, and when dragged out of the System becomes a keyboard file. This file can then be dragged onto your System Folder and the System automatically installs it into the System suitcase (see Figure 8.8). If you want to create a file that acts like a keyboard file, perhaps to make it easy for other people to install, simply use ResEdit to make a new file and give it a file type of kfil and a creator of movr.

**Figure 8.8**

If you don't want to launch ResEdit to get rid of keyboard layouts, you can just drag them out of the System suitcase in the Finder.

Keyboards Overview

So we've discovered that the KCHR resource can be used for a wide variety of purposes. As this chapter discusses, it can be used to set up keyboard layouts that may be more comfortable to use. You can make your keyboard more efficient by swapping one of the more obscure option characters you use a lot with an unmodified key that you use rarely. You can also change the way a modifier key affects the layout by switching its table with another modifier key. Whatever you try to do, the KCHR is an extremely flexible and powerful resource.

To get to the KCHR resource:

- Open a copy of the System file.
- Double-click the KCHR icon to open the resource picker.
- Double-click the desired layout to open it in the KCHR editor.

To find a key on the character grid:

- Hold down the key on the keyboard.
- The character hilites on the ASCII character grid and on the keyboard diagram.

To change one key to another:

- Find the key you want to replace on the keyboard diagram.
- Find the character you want to replace it with on the ASCII character grid.
- Drag the character from the ASCII character grid to the key on the keyboard diagram.

To reassign a modifier key:

- In the KCHR editor, hold down the modifier key you want to remap.
- Click the name of the table in the table list to which you want to remap it.

To edit a dead key:

- Type the dead key (for example, Option-E).
- Or, choose Edit Dead Key from the KCHR menu and select the dead key you want to edit.

To enable the Keyboard menu:

- Open the itlc resource from the System file in its template editor.
- Set the Always Show Icon button to 1 (true).

Safety Tips—What Not to Do!

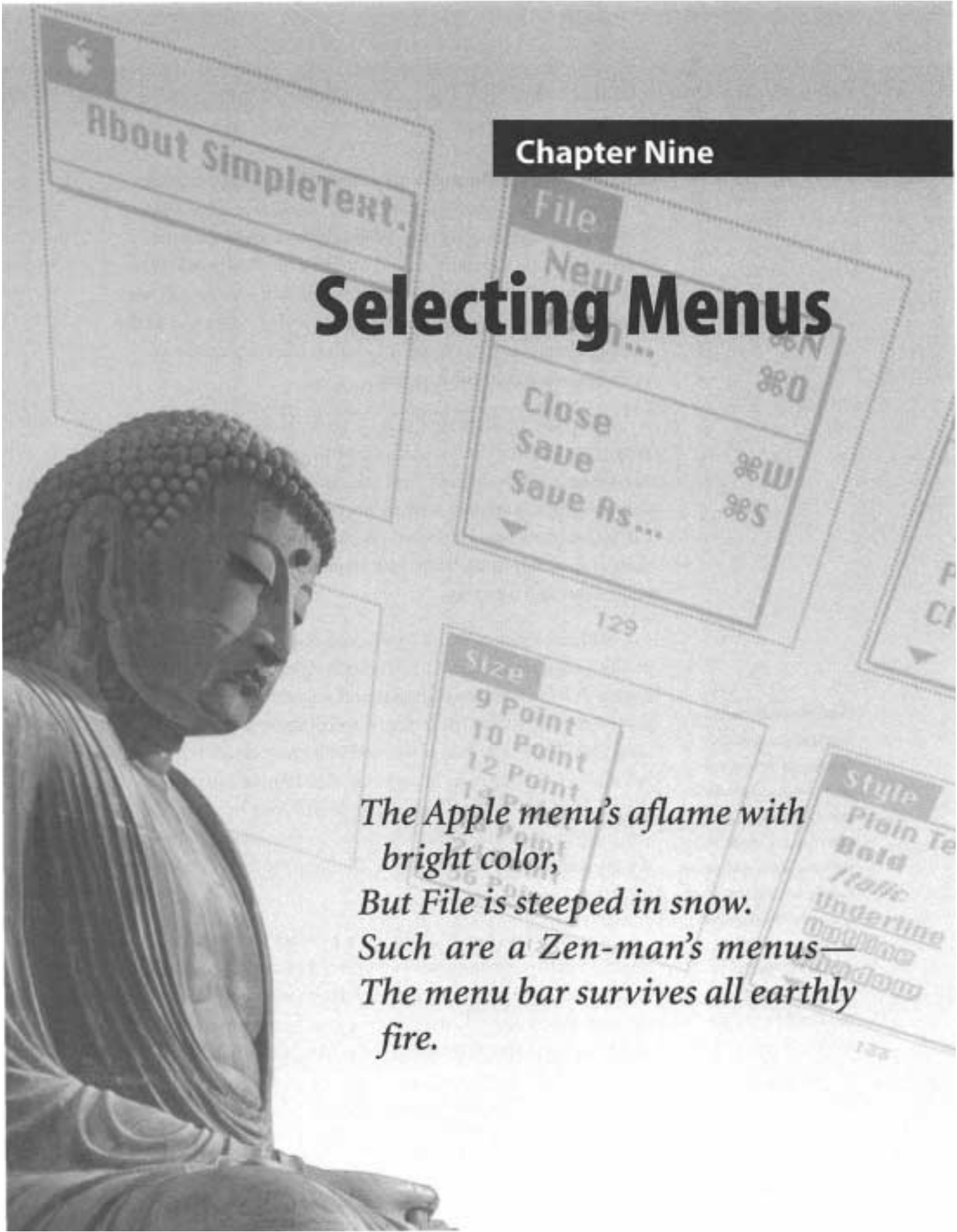
- Don't use the Uncouple Modifier Keys command.

The Buddha's Guide to ResEdit Enlightenment

- Can you design your own personal keyboard layout that's more efficient for the way you type?
- Can you make the Escape key into a modifier?
- Can you remap your power key?
- What is 16 × 16?
- If the script code for Thailand is 17, what is the range of KCHR resources assigned to it?
- What is the itlb resource for?
- Where are all the different Apple keyboard layouts stored?

Zen and the Art of Resource Editing

Selecting Menus



*The Apple menu's aflame with
bright color,
But File is steeped in snow.
Such are a Zen-man's menus—
The menu bar survives all earthly
fire.*

Menus impose structure on what might otherwise be a chaotic universe. They're organized, consistent, simple—just like a Mac should be. Rather than leaving programmers to clutter up the screen with little icon buttons and toolbars, the Mac provides an easy way to tuck commands away in the menu bar. Although some developers have taken their own initiative and gone the way of the toolbar, most Mac users recognize this unnecessary clutter as more annoying than constructive.

Every Macintosh programmer is strongly encouraged to support at least the three standard menus—Apple, File, and Edit. Apple has carefully defined these standard menus and their contents with commands such as About, Open, Save, Print, Quit, Copy, Cut, Paste, and Undo—all so Mac programmers can use these commands consistently with standard Command-key equivalents and functions.

And these menus are, of course, defined in simple resources, so all a programmer needs to do is create the names and commands that the menus will display and implement, and then hand that off to the System. The look and feel of the menus is defined and created by the System and stored in more complex resources not meant to be edited by anyone outside of Apple. But more to the point: This way, the programmer doesn't have to worry about writing code to actually draw the menu on the screen or blinking the selected item or drawing separator lines; the System takes care of all that.

Despite all this happy consistency and straightforward implementation, most menus you see are just boring columns of text, and some are poorly organized with inappropriate Command-key equivalents. Have you ever seen a program in which the developer doesn't follow the Mac standard and put

Okay—the newest version of PageMaker, 6.0, fixes the Command-W problem, now closing windows. But while it drove Derrick crazy that it didn't close the window with Command-W, it now drives Hans bonkers when he tries to change the view of the page he's working on to Fit in Window and now the document just closes. So, ResEdit to the rescue once again!

ellipsis dots after menu commands that show a dialog? How about a program that doesn't use the Macintosh standard Command-key equivalents? (It used to frustrate the bejeebers out of me that pressing Command-W in PageMaker didn't close the window.) How about a program that doesn't provide a Command-key equivalent for a command you use frequently? All these things can make using your Mac a slightly more tiresome experience. But take heart, because we will, of course, show you how to use the magic of ResEdit to solve these annoyances.

Picking a MENU

Most programs (except the Finder and nonstandard ones such as most Microsoft applications) use MENU resources to define the pull-down menus you see. Because ResEdit provides a nice editor for MENUs, you can easily edit the menus from virtually any program to suit your preferences.

To gain some experience editing MENU resources, open a copy of SimpleText (or TeachText) with ResEdit. If you don't have either of these, you can follow along with some other program. However, copies of SimpleText and its predecessor TeachText are on the CD-ROM that comes with this book and are ubiquitous on disks from Apple and most third-party software, so you should be able to find one without too much trouble.

Opening the MENU picker from SimpleText provides you with a nice view of all the MENUs in the resource fork of the program (see Figure 9.1). Yours may look a little different if you have a different version of SimpleText, but it will be close enough.

The Buddha's Revelations

Although the MENU resource contains the contents of the menu, the MDEF resource controls what the menus actually look like.

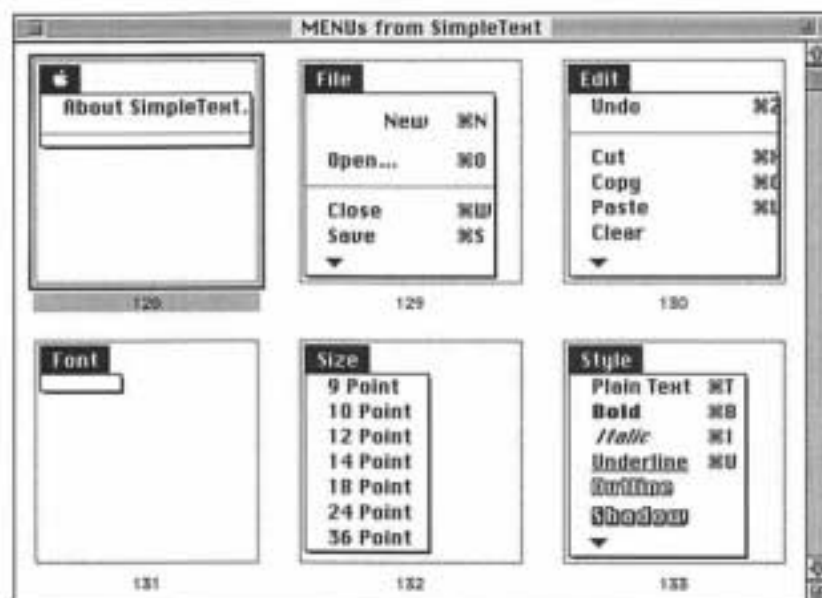


Figure 9.1
ResEdit's MENU picker,
showing the MENU resources
in SimpleText.

The MENU Editor

ResEdit does a lot of things to make editing MENU easy. Open up MENU 129 by double-clicking it. This is SimpleText's File menu (see Figure 9.2).

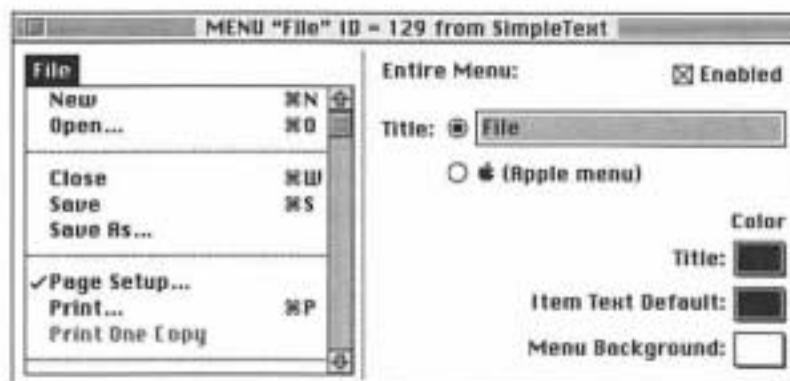


Figure 9.2
The File MENU from
SimpleText, opened in the
MENU editor.

The left side of the editor shows the MENU resource that you are editing, and you can just click components of the MENU to edit them. You can also click and drag menu items around in the

editor to reorder them, but you shouldn't do that, or none of the commands will behave the way they should. When you first open a MENU resource, the MENU's title is selected, and as a result the rest of the editor acts on the menu title. When you select a menu item, the editor changes the details of that item.

The most obvious element of the editor is the name of the menu. You can tell ResEdit to change the name of the menu or to register this menu as the Apple menu. Usually, you won't change either of these things, because this is for use by programmers. One thing you can change, however, are the colors used to display this item (only if you're on a color machine, of course). The lower right portion of the editor contains a set of color palettes (exposed when you click the mouse button in the rectangle) that set the color for the name of the menu, default color for any menu item, and background of the entire menu. If you do change the colors in the menu, ResEdit tells you that it is going to add an mctb resource to the file.

An mctb is a resource that describes the color information for a particular menu. If you try to look at the mctb resource, ResEdit puts you into the MENU editor for that particular menu.

You'll notice that the upper right corner of the editor has an Enabled check box. This determines whether the entire menu is enabled (when it is disabled, it can be pulled down, but all the items in it will be grayed out). While this is tempting to play with, the state of a menu when it's displayed is usually dictated by the code in the program, and so checking this doesn't usually do anything.

Clicking a particular item in the menu on the left side of the editor changes the right side of the editor to reflect the fact that you're editing a menu item, and not a menu, reinforcing this by changing the Entire Menu text in the editor to Selected Item

The Mac operates menu commands by their order in the menu, not the actual wording. Switching New to Open and vice versa simply causes choosing New to ask you which file you want to open. This is why it's not a good idea to rearrange your menu items—unless you're playing a cruel practical joke on the inexperienced Mac user in your office.

ResEdit also makes this a separator line if you enter a hyphen as the text of the menu item.

Zen and the Art of Resource Editing

(see Figure 9.3). Once again, you can change the text of the menu item or you can set this menu item to be a separator line with the radio button.

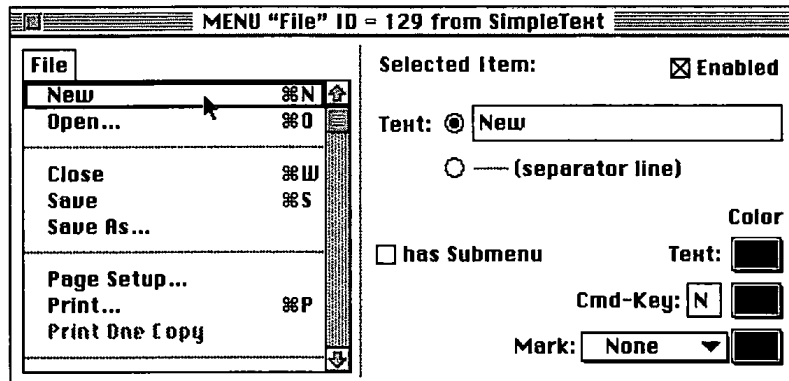


Figure 9.3

The File MENU from SimpleText, with the New command menu item selected in the MENU editor.

One of the most useful things you can do to edit a particular menu item is in the lower right. The Cmd-Key field actually lets you edit the Command-key equivalent used to activate this particular menu item. Command keys can speed you up a lot if you're using a program in which you spend most of the time on the keyboard; rather than lifting your hand off the keyboard and moving the mouse to the appropriate menu item, one quick key sequence does the same thing. If a program has a hard-to-remember Command-key, or if it doesn't have one at all, you can modify that by typing the letter you wish to use into the field.

You can, of course, use any letter you want for the Command-key equivalent, but there are some general things to remember when you're playing with this. First of all, Apple has designated certain letters to always be associated with particular menu items. Thus, Command-O is almost always Open and Command-Q is almost always Quit. Here are some common menu items and their standard Command-key equivalents:

The Buddha's Revelations

Not only will Command-keys be invisible when they're on a menu item with a submenu, but they won't function at all.

Menu Command	Standard Command-Key Equivalent
New	N
Open	O
Close	W
Save	S
Print	P
Quit	Q
Undo	Z
Cut	X
Copy	C
Paste	V
Select All	A
Find	F
Find Again	G

If you use the same letter for two different menu items in the same program, the System follows a simple rule to determine which menu item gets activated when you press the Command-key equivalent: The rightmost menu item (or the lowest menu item if they're in the same menu) is the one that is activated. In general, of course, it's not wise to set the same Command-key for two different menu items.

Below the Cmd-Key field is a pop-up menu that enables you to set the mark next to a menu item. Marks are things such as check marks, diamonds, and various other symbols. The pop-up menu enables you to choose among the common ones and provides a facility for adding your own. Unfortunately, the mark next to a menu item is usually set on the fly by the program, so changing it here usually won't have any effect.

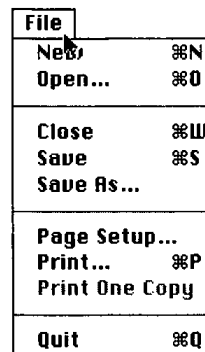
Of course, you could always use something such as **QuickKeys** or **NowMenus** to add Command-key equivalents to your favorite menu item, but real hackers do it with **ResEdit**. Using **ResEdit** also has the advantage of not needing a large and possibly buggy System extension to have access to your customized commands.

For each of these items, you can set the color that's used when the menu item is displayed. Once again, if you change one of these, ResEdit adds a mctb resource if there isn't one already defined for this particular menu.

By the way, double-clicking a menu item that has a submenu will bring that MENU resource up in its editor if you need quick access.

For the programmer, there is a check box that dictates whether the menu item has a submenu (also known as a hierarchical menu). If this box is checked, a field appears asking for the ID of the MENU resource that should be used for the submenu. While you can add submenus to existing commands, the programmer needs to add support for them, so you'll see the arrow indicating that you have a submenu, but no submenu appears unless the menu item is already supposed to have one.

As you're playing with the menu and menu items, you'll probably notice that the left side of the editor isn't the only place to see what the menu will look like in practice. A menu gets added to the menu bar at the end of all of ResEdit's menus that is the menu you're working on. You can tell it's not a real menu by the rectangle around the name of the menu (see Figure 9.4).



File	
New	⌘N
Open...	⌘O
Close	⌘W
Save	⌘S
Save As...	
Page Setup...	
Print...	⌘P
Print One Copy	
Quit	⌘Q

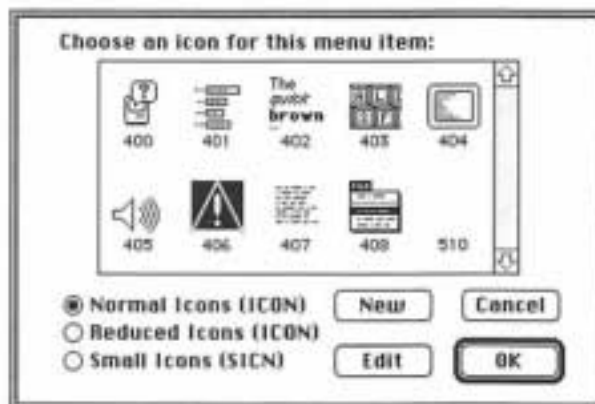
Figure 9.4

ResEdit provides a sample of the MENU you're working on after all of its menus.

It's only appropriate that the MENU editor should have some additional menus, but they don't do a whole lot. The Style menu (which is only active when you're working with a menu item, not the entire menu) enables you to set the text style of the menu item.

The MENU menu does have some interesting commands in it. One of the commands we found most useful when working on the book was the Remove Colors command. This enabled us to undo all the damage we had done playing with the color palettes, and set the MENU back to its boring black-and-white.

If you have a menu item selected, you can choose to attach an icon to it with the Choose Icon command. This is kind of cool at first, but we've rarely see it done with much in the way of aesthetic consideration. When you choose this command, you are presented with a dialog that enables you to choose an icon from within the same file (see Figure 9.5). The icons it uses are ICON resources (at normal size or scaled to one-quarter the original size) or SICN resources, which you are no doubt an expert in after reading the icons section. If you're using an ICON resource and have a cicon resource with the same ID, the System uses the color icon on color machines. To show the different types of icons, use the radio buttons at the bottom of the dialog. You can even create a new icon (or edit an existing one) from within the dialog by clicking the New or Edit buttons, respectively.



Actually, ResEdit won't let you use any icon in the file; it has to have an ID number between 257 and 511. And if your menu item has a reduced ICON or SICN, you can't have a Command-key associated with it, so you have to use normal icons if you want Command-keys for a particular menu item.

Figure 9.5

The MENU editor enables you to attach ICON resources to a particular menu item. Unfortunately, SimpleText doesn't have any ICON resources, so the dialog is empty.

Of course, you can also remove an icon by using the Remove Icon command in the MENU menu. This is handy for reducing the menu item back to its normal state quickly.

The Buddha's Revelations

If you're curious, Balloon Help for menu items is stored in the `hmnu` resource.

Not satisfied with the colors you have on your palette? If you choose Use Color Picker from the menu, you'll be presented with the full color picker whenever you attempt to set the color for a menu element. This gives you a much wider range for attractive menu-coloring possibilities.

Lastly, don't worry about the Edit Menu & MDEF ID command. This associates the MENU resource with a particular ID for Macintosh Toolbox routines. Unless you're a programmer, you'll never have to worry about this.

Now that you've seen how to edit MENU resources, you can unleash your newfound powers on your unsuspecting programs. Change Command-keys to make life easier for yourself. Add icons to make menu commands more meaningful to new users (or yourself). Apply garish colors to your heart's content, or subtle ones for a nice effect. Remember to work on a copy, or at least remember where the Remove commands are in the MENU menu.

Except...

One place that is desperately needing some Command-keys is System 7's Finder. Until System 7.5, such obvious things like Make Alias didn't have Command-key equivalents. Can you imagine?!

So, you go into the Finder with ResEdit and come to a screeching halt. There are no MENU resources! What's going on here? Surely the System itself uses MENU resources?

Well, we have a solution for you! But you're going to have to wait a bit to hear about it, or you can skip ahead to Chapter 12, because changing the `fmnu` resource has absolutely nothing to do with the MENU resource or its editor.

Menus Overview

Menus are a constant in the Mac universe, saving us from screenfuls of hideous icon buttons (well, most of the time). However, that doesn't mean they're always perfect. Armed with your newfound knowledge, you can adjust all your menus to make you more efficient, add a more colorful tint to your Mac, or just play jokes on your coworkers. Feel free to explore all the possibilities!

To get to the MENU resource:

- Open the program you want to alter using ResEdit.
- Double-click the MENU icon.

To change an aspect of the entire menu:

- In the MENU resource, open the menu you want to modify.
- Make sure that the editor says Entire Menu.
- If you want to change the name of the menu, type the new name into the Title field.
- If you want to make this menu the Apple menu, click the button next to the Apple symbol.
- To change the various colors of the menu, hold the mouse button down in the square next to the color you wish to change. Once the color palette appears, choose the color you would like.

To change a menu item:

- Open the MENU resource you would like to change.
- Click the menu item once that you want to alter.
- To change the text, type the new text into the Text field.
- To make this item into a separator line, click the button next to the dotted line.

- To change or add a Command-key, type the new Command-key into the Cmd-Key field.
- To attach a submenu, click the Has Submenu box and then type the ID number of the menu that is to be the submenu.
- To change a color attribute, hold down the mouse button over the square next to the appropriate area. When the palette appears, choose the color you would like to use.
- To add a mark to a menu item, hold down the mouse button on the Mark pop-up menu and choose the mark you want to add.

To add an icon to a menu:

- Go to the menu item to which you want to attach an icon.
- Select Choose Icon from the MENU menu.
- Select the icon you want to attach.
- Use New or Edit to create a new icon or edit an existing one.

Safety Tips—What Not to Do!

- Switching around the position of any menu item or adding separators can cause unexpected results. Programs implement a command not by that menu item's ID or name, but by its relative position in the menu. Changing the position of a menu item doesn't change the position of the command that is activated by that position.

The Buddha's Guide to ResEdit Enlightenment

- What happens to Command-keys on menu items with submenus?
- Why don't Microsoft applications have MENU resources?
- Which menu in the System is the application menu?
- Where is the Keyboard menu?
- What menus are in the Color control panel?
- What is the only System menu item that doesn't have balloon help?
- What does the hmnu resource store?
- What do you think the MDEF resource is for?

Zen and the Art of Resource Editing

Chapter Ten

Dialog Boxing



The void collapsed upon the earth,
Stars, burning, shoot across Iron Mountain.
Turn a somersault and brush past?

OK

Many Mac programs have certain common resources, simply because they are Macintosh applications. Menus and icons, covered in Chapters 9 and 3 respectively, are two of these. Another common type is the dialog box. The generic term “dialog box” actually covers two concepts: dialogs and alert boxes. Dialogs request information from you, as in the standard open file dialog you see when you instruct an application to open a file (see Figure 10.1). Alerts, on the other hand, simply provide you with information, as in when you try to give a file a name longer than 31 characters in the Finder (see Figure 10.2). In general, dialogs are asking for some type of input or decision, and alerts couldn’t care less about what you think of a situation.

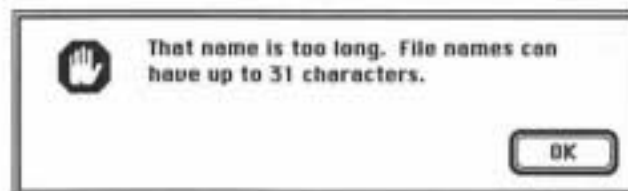
Figure 10.1

Two types of dialog boxes are used on the Macintosh: dialogs...



Figure 10.2

...and alerts.



Where Do Dialogs Come From?

Dialog boxes and alert boxes both have certain standard components. Let’s look at each of these items in more depth. Open a copy of the System file and double-click the DITL icon to open the DITL picker. You’ll find that the System has a whole lot of

dialog box resources. If you open any of these, you see that each resource defines the layout of a whole dialog box. For the sake of following the text, open DITL -4000—again the standard open file dialog box, only this time from the “inside” (see Figure 10.3).

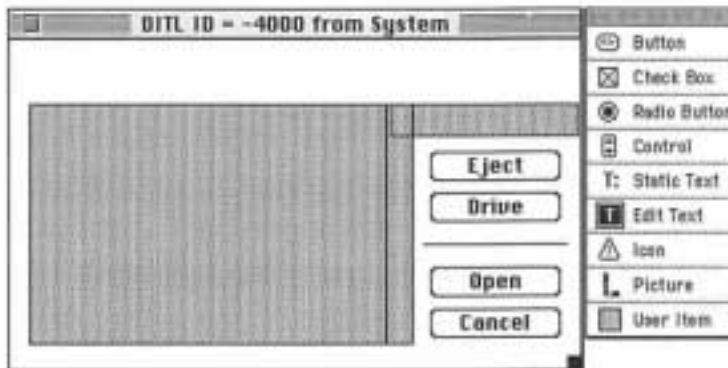


Figure 10.3

The DITL editor in ResEdit showing DITL -4000 from the System file, more commonly known as the open file dialog box.

ResEdit displays a window with the elements of the dialog box represented by boxes and labels. Next to this window is a tool palette of dialog box parts. These parts can each be added by clicking and dragging these elements from this palette onto the dialog editor window. Let's take a quick look at each of these parts:

- *Push buttons* are the rounded rectangles within a dialog box that usually contain words such as OK or Cancel. Pressing one usually causes the dialog to disappear, though some programs take you into a new dialog box when one of these is clicked.
- *Check boxes* are usually used to indicate an option that can be toggled on and off. The user doesn't have to do anything with a check box; it can just sit there.
- *Radio buttons* require a user to make a selection from a range of choices, each with its own radio button. Unlike with check boxes, the user does not have the option of leaving them all off or selecting more than one. With a group of radio buttons, only one may be selected at a time.

The Buddha's Revelations

Unfortunately, you can't usually change the default state of a check box or radio button, because this is encoded in the program itself.

If there are two or more groups of radio buttons, one in each group must be selected.

- *Control* refers to the scroll bars used in a long list.
- *Static text* is the text that is permanent for the dialog. At the beginning of the chapter you saw an alert box; the text in this alert is static text. It can be changed in ResEdit, but it cannot be altered in any way within the dialog.
- *Editable text* is the text that the user types into the dialog box. The most well-known example of this is in the save dialog box of every Macintosh application. When you save a document the first time, you are asked to provide a name. The place where you type this name is an editable text field.
- *Icons* can be used in a dialog. The System uses ICON resources in its dialogs. However, as you know, setting up a cicon with the same ID as an ICON resource causes the color version to be displayed. cicons can be up to 64×64 , but the ICON resource is limited to 32×32 .
- *PICT* is a picture resource. The major benefit of a PICT rather than an ICON is that a PICT is not limited to 32×32 pixels.
- *User items* are items that the programmer specifies. These are objects not supported separately by the Dialog Manager within the Toolbox. Some examples of user items include pop-up menus, thermometers (which indicate progress of an action), and userText, which can be used to create different font styles within a dialog box. Since this item is for programmers, we won't discuss it here.

Editing the Dialog Items

Rather than playing with the standard open file dialog box, let's create a new dialog box to practice with. Close DITL -4000, and with the DITL picker still open, choose Create New Resource

from the Resource menu. ResEdit gives you a blank dialog box with the DITL tool palette, which looks like what you see in Figure 10.4.

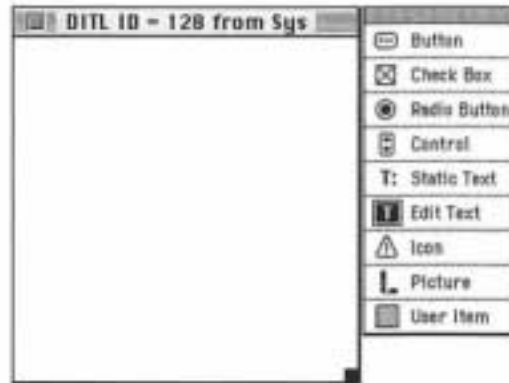


Figure 10.4
A blank DITL resource
demanding your creativity.

To add an item to a dialog box, click and drag the desired part from the tool palette. You may want to experiment to see how items with odd sizes (such as really big buttons) are presented within the dialog. For now, create one of each item in your dialog box (see Figure 10.5).

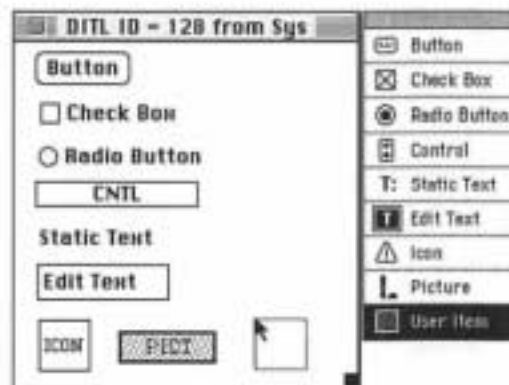
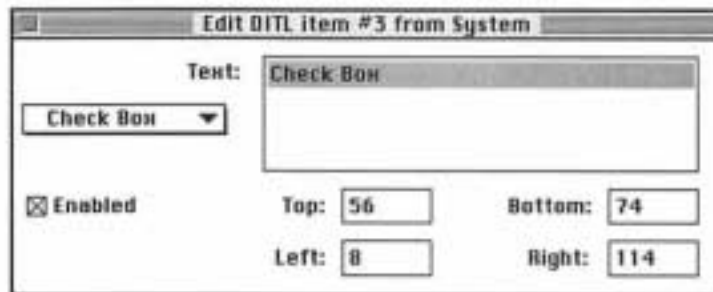


Figure 10.5
It doesn't look very
functional, but this DITL is a
dialog box that you have
created.

Right now, your dialog doesn't say much. Your check box has the words Check Box next to it, your radio button has the words Radio Button next to it, and so on. Double-click the Check Box item to edit it. You should get something that looks like Figure 10.6.

Figure 10.6

The editor for an individual dialog item; in this case, a check box.



Let's look at each of these parts. There is a pop-up menu on the left that enables you to change what kind of item it is. For instance, let's say that you finished your dialog box, and then realized that you put a check box where you wanted a radio button. Just open the pop-up menu and choose Radio Button. Your item will suddenly become a radio button.

The title of this window indicates that this is DITL item #3. What does this mean? We'll discuss this in more depth in the section about the menu items in the DITL editor. For now, just realize that each item in a dialog has a number associated with it so that it can be identified by the programmer.

The Text box enables you to change the text that appears next to the item, if applicable. This applies to check boxes, radio buttons, static text, push buttons, and editable text. In the first four, this text is permanent; it can't be changed except in ResEdit. In editable text fields, however, this is the default text of the field; it can be changed by the person using the application. For example, when you save an item and are asked what to name the file, most programs already have a name in this field. You just type over it.

The Enabled check box below the pop-up menu enables you to determine whether an item is enabled, just as it did in the MENU editor you saw in Chapter 9. If this is not checked, the user sees a grayed-out button, check box, radio button, or text entry field, which cannot register mouse clicks or accept text.

The four fields below the Text field tell you the dimensions of the item. Top and Bottom indicate what pixel rows the item spans. Similarly, Left and Right tell you which pixel columns the item's edges are on. Note that these are relative to the dialog box itself, not to the screen. Thus, in Figure 10.6, the top edge of the check box is 56 pixels from the top of the dialog box, and the left edge of the check box is 8 pixels from the left side of the dialog box. (The 0 position is defined as either the topmost row or the leftmost edge of the dialog).

This editor works fine for text-based items, but some different information is needed for things such as icons and pictures.

Close the Check Box editor and double-click the icon item in your dialog to see what its editor looks like (see Figure 10.7).

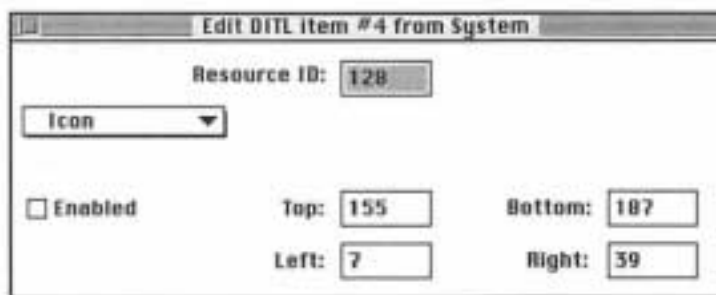


Figure 10.7

Icons and PICTs have a slightly different editor than other dialog items.

This window is almost the same as the check box editor you saw a moment ago, but it has one important difference. In the check box item editor, we had a Text field. In this editor, we can see that the field now says Resource ID. This editor enables you to set which icon you wish to use by its ID number. To use a new icon, simply type its number. If you type an ID number and that icon does not exist, ResEdit shows you a generic icon. However, your dialog won't work properly in the program.

Notice the Enabled check box again. You might be wondering how an icon can be "enabled" or "disabled." Well, programs can be

told to look for the user clicking on the icon (or the picture, which has the same editor as this one). Some programmers might use this to bring up a short piece of info about themselves or the program.

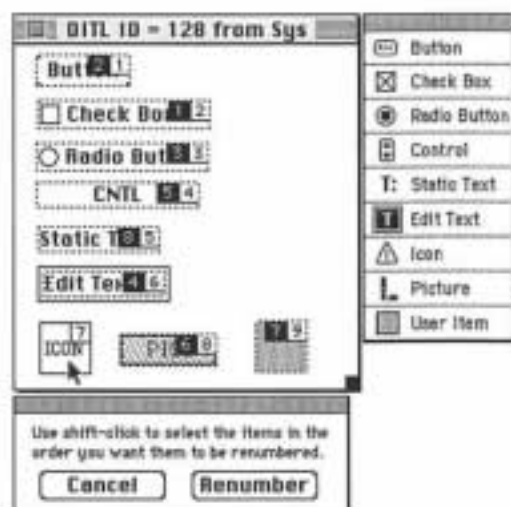
DITL Menu Items

In the DITL menu—which becomes available when you return to the dialog window—you can choose from a wide variety of commands to help you with your design.

The Renumber Items command enables the programmer to quickly renumber the dialog items. You probably won't be doing this (unless you're designing dialogs for use in something such as a HyperCard stack), because renumbering items can cause very strange results. But if you're interested in how it works, it's pretty simple. When you choose the command, ResEdit pops up a little number in each dialog item and shows you a palette that enables you to cancel the operation or put it into effect. To actually renumber the items, hold down the Shift key and click the items in the order in which they should be renumbered (remember, just look!). The first item you click is renumbered to 1 (even if it is already 1, which of course seems to have no effect). The next one is 2, and so on.

What do these numbers actually do? Basically, they refer to the placement of items in the information the programmer gets back when the dialog is dismissed. The programmer looks through the information and writes the equivalent of "If item 3 is true (for example, it's a check box that has been checked), then use option x." Thus, if you renumber the items yourself, you could cause all sorts of weird things to happen (if you manage to avoid causing the program to crash!).

When you are ready to renumber the items, click the Renumber button (see Figure 10.8).

**Figure 10.8**

You can quickly renumber your dialog items. Just remember that the first item you click is always renumbered to 1.

The Select Item command enables you to select an item by its number. Once you type the number, the corresponding item becomes selected and you can manipulate it. This is particularly handy when the programmer is touching up a full dialog and needs to find a particular item quickly.

The Show Item Numbers command, when checked, shows all the item numbers of the items in your dialog. The number appears in the upper right corner of each item. This way, you can quickly see which item is which.

The Align to Grid command constrains the items to an invisible grid. The items can be put only along the grid, aligned to one of the lines.

The Grid Settings command allows the spacing of the grid to be changed.

The Show All Items option forces ResEdit to put a rectangle around all the items in the dialog box. This rectangle is the rectangle that is defined in the item editor (pixels from the top, and so on).

The View As command enables you to look at the item in a different font. However, this doesn't show up in the actual dialog

box; it is only for your convenience. Dialog boxes are set in Chicago 12 by default. Except with user items, this is not changeable in a specific DITL resource.

The Balloon Help menu command enables you to select which Help balloon resource you wish to use. System 7 features the capacity to have help balloons in dialog boxes. Thus, a person who is unsure of what a dialog box item does could use a help balloon to gain some more insight about it (see Figure 10.9).



Figure 10.9
Balloon help from the Find dialog in the Finder.

Next to the DITL menu is the Alignment menu, which is fairly self-explanatory, enabling items in a dialog box to be lined up with one another. In order to use this, two or more items must be selected. Select more than one item by holding down the Shift key while clicking each item.

Of course, this may seem like useless info. You're probably not a programmer, so you won't be designing your own dialogs. Well, that's true, unless of course you're designing dialogs for HyperCard. Nonetheless, it's occasionally useful to know details about how Macintosh applications work. After all, everyone needs to converse with nerds at some point, right?

Aside from the warm, fuzzy feeling of knowing how dialog items work, you might like to reword a dialog so that you are more able to discern its meaning (once you figure it out the first time, of course). Alternately, you might want to reorder the contents of a

dialog; we have a friend who completely redesigned the interface of a program by moving all the irrelevant dialog items off the edges of the dialog (you'll see how to change the size of the dialog in a bit when you learn about DLOG resources) and rearranged the remaining items to be more aesthetically appealing. Furthermore, you'll be amazed at the types of practical jokes you can play on friends and family when you know some of these details.

The DLOG and ALRT Resources

DITL resources control only what appears in a dialog box. Two other resources, DLOG and ALRT, determine what type of window the dialog is displayed in and where on the screen it shows up. Figure 10.10 shows what DLOG 128 would look like for the sample dialog you saw in Figure 10.5, which we created in a copy of the System.

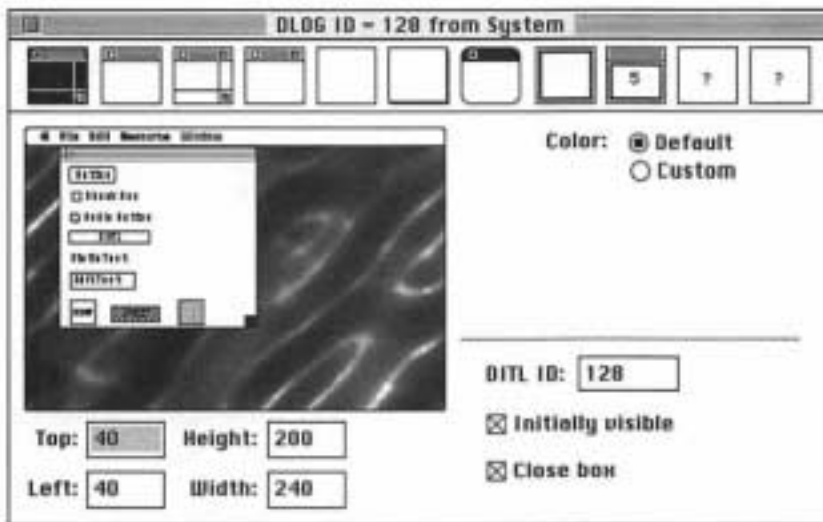


Figure 10.10
The DLOG editor showing
your DITL resource in
miniature.

The most prominent aspect of the DLOG editor is a miniature version of the Macintosh screen (complete with desktop pattern). This enables you to position your dialog box on the screen. To move it, simply click and drag it to its new location.

Zen and the Art of Resource Editing

About WDEFs. WDEFs are not resources ResEdit can edit—it can only access and use them. You can find them in various System customization extensions, but you would have to study low-level Mac programming to make your own.

Another important aspect of the DLOG editor is the row of window styles along the top of the screen. Let's look at these one by one. The first from the left is a standard window, complete with scroll bars and a grow box, but missing a zoom box in the upper right corner. The next one is a window with just the close box in the upper left corner. The third is a normal window with a title bar, close box, zoom box, grow box, and scroll bars. Next is a window with a title bar, close box, and zoom box, but missing everything else. Next is just a plain rectangle. Similar to that is the rectangle with a drop shadow, which can add a little bit of depth to your image. After that, you'll see a flat dialog with rounded corners. The next style is a rectangle with a double border around it. This is a common style for alert boxes. The last three styles can be defined by you, if you have developed a custom WDEF (Window DEFinition) resource, or if you have one you would like to use.

At the bottom left corner of the DLOG editor, you can see the measurements (in pixels) of the dialog. As before, 0 is defined as either the top of the screen or the left edge, depending on context. When you have a DLOG, you must tell the Macintosh which DITL resource it is controlling. This is the purpose of the DITL ID field. If you double-click the sample dialog box on the miniature Macintosh screen, you bring up the DITL editor for that resource. There is also an option for determining whether a dialog is visible as soon as it is called. This is useful for programmers who want to dynamically set the text or other items in a dialog; they can build it offscreen and show you the final result, rather than making the changes after the dialog is displayed (though before you'd notice it on a reasonably fast Mac). The final item is the check box for Close box. If this is selected, the window has a close box in the title bar. If not, it doesn't.

Finally, there is the ALRT resource (see Figure 10.11). This is actually very similar to the DLOG editor, with some slight

The Buddha's Revelations

Because alerts are usually informing you about something rather than asking you what you want to do, they don't have Cancel buttons.

differences. Again, it enables you to reposition the dialog box and determine its size. It does not have the option of different window styles, nor does it have the Initially visible or Close box options. It does still have a field containing the ID number of the DITL resource you are using.

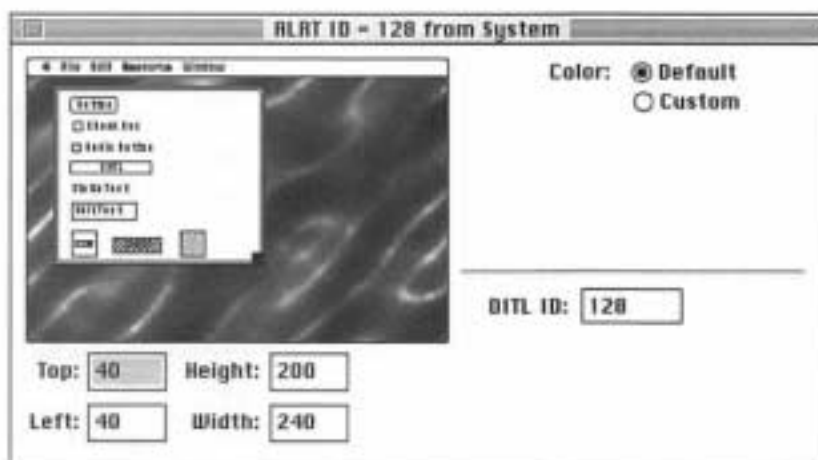


Figure 10.11

An ALRT resource showing the DITL resource you created earlier, in miniature.

In summary, DITLs are the resources that control what items are in a dialog. DLOGs and ALRTs then determine where the DITL is and what type of dialog it is: regular or alert. Although you can't add a lot of functionality to existing dialogs, you can rearrange things to suit your own tastes, and you can add things such as icons and pictures to make the dialogs more intuitive.

Numbering

Remember we said earlier that all items in a dialog have a number associated with them? This is very important. If a user clicks an item, how does that cause resource information to be fed to the program? The answer is through these numbers. By using the item numbers, a programmer can tell a program to do one thing if a radio button is "true" (selected) and another if it is "false" (not selected). The programmer does not type, "Tell me what that

radio button says.” She types in the equivalent: “Get the return from item number 6.” The program does not know that it is looking at a radio button; it tells itself only that the result is “true” or “false.” Thus, numbering is very important. When a resource is activated, all the items send information to the program, and it is up to the programmer to work with that. Even items that are not selected send back “false” messages so that the program knows that that option was not selected.

Items can send back other information as well. Editable text sends back the contents of the field. Static text sends back what it says, even though this doesn’t change while the dialog box is active.

Certain items have special requirements for numbering. One of these is the default button. Dialog boxes often have a default button (indicated by a thick, black border) that you can usually hit the Return key to activate. This button is item number 1, and it is always a good idea to have item number 1 be a button. This prevents confusing results. Similarly, item number 2 should be another button (usually Cancel). This obviously doesn’t apply to alerts.

Another important concept in numbering is with radio buttons. You may have noticed that, when you have a group of radio buttons in a dialog, you can select only one of these. This is due to the numbering of this group of radio buttons. When you assign numbers to them, you must make all the radio buttons in a group sequential and consecutive. In other words, the radio buttons, in order to work properly, must be something like 3, 4, and 5. They cannot be 3, 6, and 8, because then they will all be considered as separate groups. The Macintosh’s Dialog Manager then makes sure that one is selected at a time. If you have two groups, they must be separated by at least one other dialog item.

Color Dialogs

People with color Macs see a slightly different DLOG and ALRT editor. With color capability, a pair of radio buttons are visible. These enable you to choose between the default colors and a custom set of colors. To add your own color scheme, choose the Custom radio button. ResEdit then presents you with color palettes for a wide variety of the areas in the dialog which are fairly self-explanatory—Frame is the frame of the dialog, Content is the main area in the dialog, and so on (see Figure 10.12).

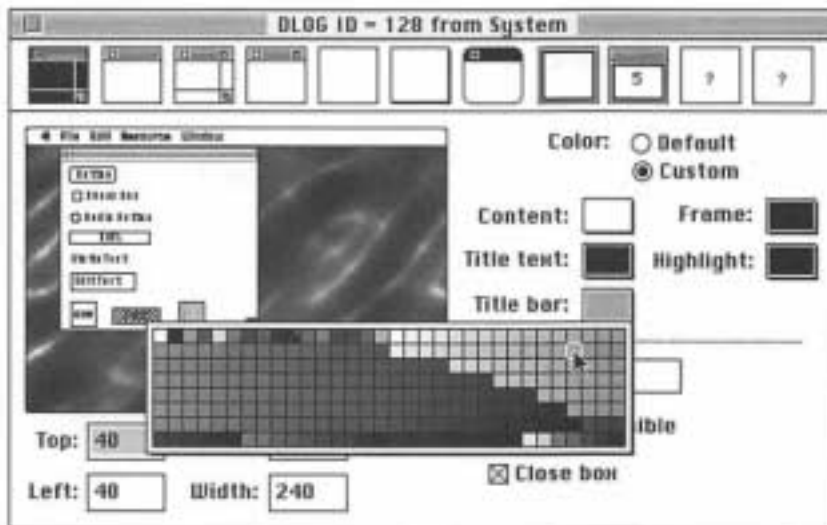


Figure 10.12

With a flick of the wrist, you can make your dialogs color.

To choose a new color, click and hold one of these boxes and a palette of colors appears. Still holding down the mouse button, move to the color you wish to use.

When you add colors, ResEdit warns you that it is about to add a new dctb. This is short for Dialog Color Table, and it represents the colors in a DLOG. The dctb has the same resource ID number as the DLOG for which it is used.

This same technique can be used for alerts as well, but ResEdit adds an actb to the file instead of a dctb.

Dialog Boxes Overview

Most people who create dialogs are programmers. However, there is nothing to stop you from editing dialogs in existing programs. Some people alter dialog boxes so that the contents say something different. Others like to put in new items that have no meaning at all, as practical jokes. However, you should always be careful about editing dialogs. Never change an existing item into another type or rearrange the numbering scheme. The program will probably get very confused if you change a radio button into a check box. With this tip in mind, you should be fairly safe.

To get to the dialog resources:

- Start ResEdit and use it to open the file you want to look at.
- Double-click the DITL resource to show all the dialog item resources.
- Double-click the DLOG resource to show all the dialog resources.
- Double-click the ALRT resource to show all the alert resources.

To change the text of an item:

- Double-click an item within the DITL.
- In the editor, type the new text into the Text area.

To change the icon or picture used in a dialog:

- Double-click the ICON or PICT resource.
- Type the ID number of the resource you want to use instead.

To renumber the items in a dialog box:

- Choose Renumber Items from the DITL menu.

- Click the items in the order (starting from 1) in which you want to put them.

To add color to a dialog box:

- Click the Custom radio button in the DLOG editor.
- ResEdit shows you a set of color boxes for various parts of the dialog.

Safety Tips—What Not to Do!

- Don't renumber dialog items. The program uses these numbers to know how to behave as each item is clicked, and you are likely to get unpredictable errors.

The Buddha's Guide to ResEdit Enlightenment

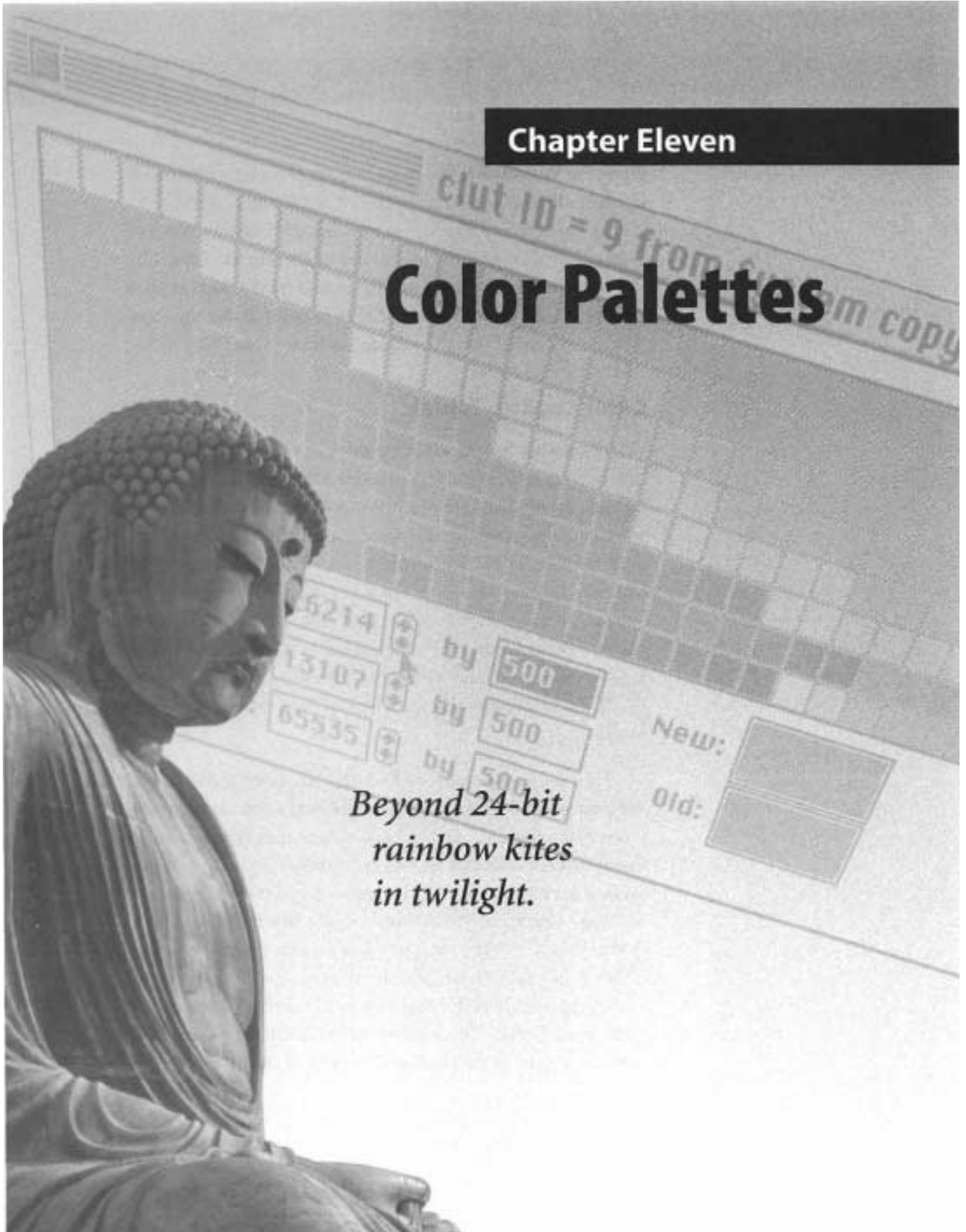
- Can you make your standard file dialogs (for opening and saving files) wider so that you can see the full filenames?
- Can you make a check box come up checked by default?
- Can you make a different push button come up as the default?
- Can you find the secret About the Finder dialog?
- Why don't alerts have Cancel buttons?
- How would you design a dialog that you could use from an AppleScript? (Of course, you could show it from an AppleScript only if you had DialogRunner, which just happens to come with the book *Tao of AppleScript*.)
- If you see a ^1 in the text of a dialog item, what does it mean?
- Can you make your standard file dialogs come up upside down?

Zen and the Art of Resource Editing

Chapter Eleven

Color Palettes

*Beyond 24-bit
rainbow kites
in twilight.*



When the Macintosh II first came out, Apple proudly advertised that it could display a dizzying 16,777,216 discrete colors! The human eye can only distinguish approximately every 1,000th of these colors. This gave the Mac photo-quality display capability. But when people got their first Mac IIs and tried to look at photo-realistic images on their monitor, they discovered that they could look at only 256 of these 16 million colors at a time. Why is this?

Color Lookup Tables

The answer has something to do with a resource called the clut, or color look-up table. You see, the first generation (and even second generation) color Macintoshes couldn't handle the huge amount of data needed to display those 16 million colors all at once, because for every pixel on the screen, the Mac must keep track of a binary value for its color. For the Mac to be able to animate an image such as the smooth movement of the arrow cursor moving around the screen, it has to calculate a new value for every pixel and then write those values to a huge table of data that defines the image you see. The Mac does this 60 times every second.

For a somewhat standard 13-inch Mac screen of 640 pixels by 480 pixels, that is a total of 307,200 pixels to calculate and track. If you are running a Mac in black-and-white, then there are only two values to track—on and off, or 1 bit of data—for each of those pixels. That's 307,200 bits. This is easy and pretty fast. If you are running in the standard 256-colors mode, that translates into 1 byte of data, or 8 bits, to keep track of for every pixel—or approximately 2,457,600 bits. To show the millions of colors all at once, you need three layers (for Red, Green, and Blue) of 8 bits each to track for each pixel. This total is a huge 7,372,800 bits, or 24 times as much data as on the black-and-white Mac. Because of the

amount of data involved, Apple created the color look-up table—a kind of color shortcut that turned the millions of colors into a mere 256.

Because Apple initially limited the number of colors visible at any time to 256, it defined a basic color palette that it calls the System palette. This is used by the System, Finder, and all applications unless they override it with their own palette. It is designed to cover the entire spectrum of color so that you can create very colorful images despite the limitation that was imposed by processor power. You have seen this System palette when you've used the color pop-up palette in the icon editor. It is stored in a clut resource in the Mac's ROMs (see Figure 11.1). When the Mac tries to display an image that uses colors that are not in the System clut or the application's custom clut, it uses a series of rules to figure out which color in the clut is closest to the color in the image.

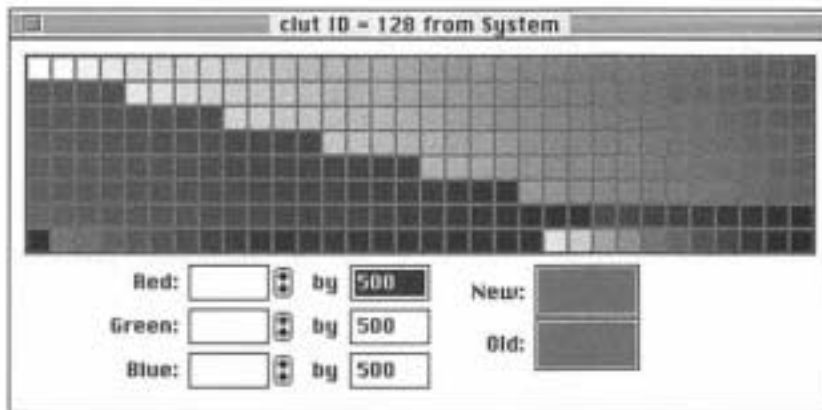


Figure 11.1

The System palette from the ROMs. This is used as the source of color on your Mac unless a program implements a different one.

Some programs use a custom color table for creating smoother blends of color through a narrower range. This includes many graphic applications that want to create sharper images or more accurate image approximations, or to simulate colors used by various printing technologies. Many games and screen savers also use custom color palettes for these same reasons.

The clut Resource Editor

So what can you do with a clut resource? Well, let's say that you want to use your own palette of colors for editing icons in ResEdit, or you want to modify the colors in your favorite game. Once again, ResEdit makes this simple with a powerful editor.

Because exploring the various System palettes (which cover a variety of bit-depths as well as full-color or grayscale options) is a bit too confusing, let's start by creating our own color palette for using while editing icons in ResEdit. To do this, we will add a new clut resource to ResEdit's Preferences file. You can add resources to ResEdit by editing a copy of ResEdit, of course, or you can add them to the ResEdit Preferences file (which is the preferable method). Resources in the Preferences file are loaded into RAM by ResEdit (see Figure 11.2).

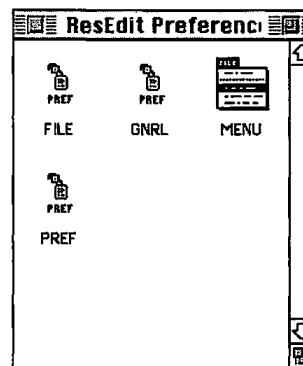


Figure 11.2

The resources in the default version of ResEdit Preferences. All these resources (and any you add) are loaded into ResEdit when it starts.

Choose Create New Resource from the Resource Menu (or press Command-K). Type clut (see Figure 11.3).

When you click OK, ResEdit adds a new clut resource with the default ID of 128 and opens it so that you can edit it (see Figure 11.4).

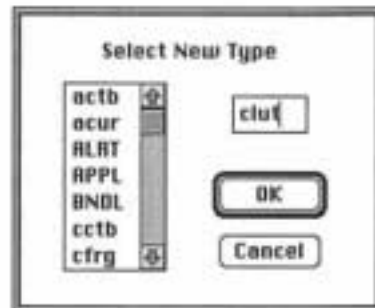


Figure 11.3
Creating a new clut resource.

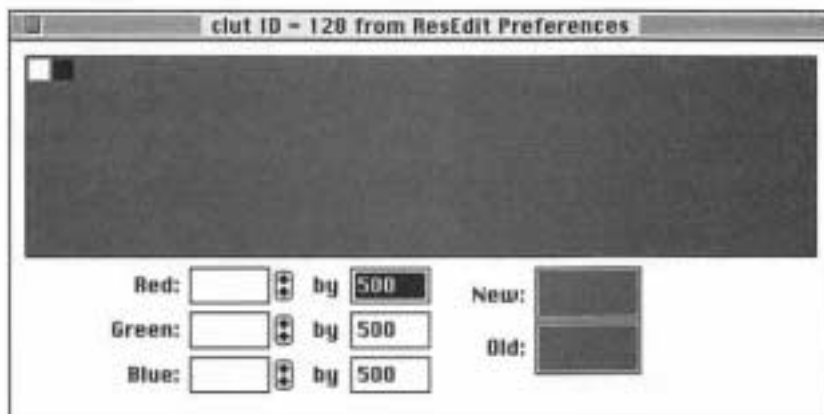


Figure 11.4
A blank clut resource doesn't have much in the way of color, but it does have 256 spaces for you to add your own.

The clut editor is made up of three parts. The large gray area can hold up to 256 little squares, each one representing one of the colors in the palette. At the bottom are controls for seeing and changing the numeric value of a selected color. To the bottom left are the component Red, Green, and Blue values for a color that can be changed up and down with the small arrow controls next to the fields. The by fields determine the amount of change that occurs in the corresponding fields when the arrow buttons are clicked.

The Red, Green, and Blue values have a range from 0 (which is black) to 65535 (which is the brightest that component of color can get). Setting all three color components to 65535 makes the color white.

Zen and the Art of Resource Editing

To the bottom right of the editor are before (Old) and after (New) samples of the color being edited. When you change a color, the New box displays the new color.

Every time you create a new clut resource, it already has two colors in it: black and white. You can modify these if you want, but it is always useful to have black and white in your color palette, so we recommend you leave them alone.

Add a new color by pressing Command-K for the Insert New Color command in the Resource menu. ResEdit adds another little square in the palette area of the editor. With it selected, you can use the controls at the bottom to change it from the default black to something else. Try typing 20000 in the Red value field, 30000 in the Green value field, and 10000 in the Blue value field. Pretty, huh?

Editing colors this way can be slow and annoying. You've probably seen the standard Apple color-picking dialog and are wondering why we can't use that instead. Well, just double-click the color in the palette and up it pops (see Figure 11.5).

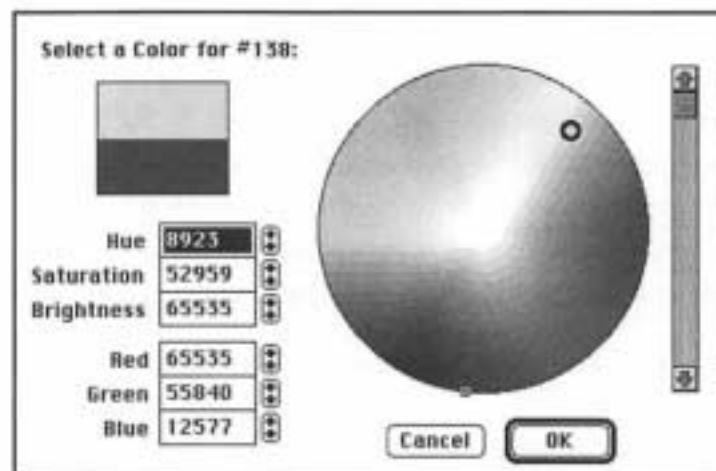


Figure 11.5

In case you don't know the RGB values of a specific color, you can use the standard color wheel (this is the color wheel from System 7.5.1).

You should find this a tad faster and more intuitive. You can add up to 256 colors to the clut. For now, add a few more colors before we close the editor and use the new palette in the icon editor.

Once you've added colors to your clut resource, close the editor. Before we close the ResEdit Preferences file, let's name our clut resource. With your new clut resource selected in the clut picker, press Command-I to Get Info on the resource. Give the resource a name such as My Wonder Palette and close the Get Info window. You should now quit ResEdit and save your changes to ResEdit Preferences. The next time you run ResEdit, you'll be able to use this new palette when you're editing any color resources.

Try opening anything that has color icons in it and open one of them in the editor. When you go to the Color menu, you'll see your wonder palette as one of the options. Select it and you'll be able to click in the color swatches in the bottom left of the editor to see and use your new palette. Exciting, huh!

Color Palette Cycling

In general, color palettes are behind-the-scenes types of things that are pointless to modify and far from exciting. Well, there is one area in which color palettes are slightly more fun. Let's say you are viewing a nice image, and all of a sudden the palette is modified on the fly and the colors in the image are just as suddenly totally screwed up. Sound like a bad acid trip?! No, but many screen savers use this technique to animate an image without having to actually move a thing.

At this point, we'd like you to run a little program that is included on the CD-ROM that comes with this book and is distributed freely wide and far. Go into the Text Examples folder on the CD-ROM and double-click Demo Flowfazer™ (see Figure 11.6).

The Buddha's Revelations

Although lots of screen savers have modules that animate cluts, some create their color palettes on the fly, so you can't edit them with ResEdit.

Zen and the Art of Resource Editing

Figure 11.6

The Demo Flowfazer icon.
This software was produced
by rock musician Todd
Rundgren.



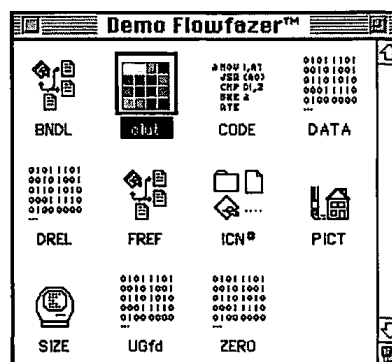
To run this app, you need a color-capable Mac running in 8-bit (256 colors). If it just beeps and quits, then it probably isn't in a color mode it can run in, or it can't get enough RAM to run. You can always just read the text and pretend to be following along.

First, Demo Flowfazer draws an interesting pattern on the screen (this may take a minute depending on what type of Mac you have). Once it has drawn the entire screen, it starts moving. Well, the trick here is that it isn't really moving at all, but just reassigning the colors in its palette to a new position one after another, over and over. This creates the flowing animation with very little work for the computer. You can click the mouse to escape anytime.

So, let's go look in Demo Flowfazer's resources and see what we find. Copy the Demo Flowfazer application to your hard drive. Drag and drop it onto ResEdit to open it quickly (see Figure 11.7). (You can, of course, run ResEdit and then choose Open from the File menu.)

Figure 11.7

One of the resources in Demo Flowfazer is the clut resource, which controls the animating colors in the program.



Open the clut resource picker and open its one clut resource, ID 0 (see Figure 11.8).

The colors in this resource should look familiar. They make up the animated pattern you saw flowing around your screen. By

starting in the first row and going from left to right, you can get an idea of how this color table is used in the flowing image to create various contrasting edges and flashing patterns. Let's modify this to get a better feel for this interaction. We'll just change this copy right now—you can always copy it off the CD-ROM again to get the original.

Click in the first white square in the first row to select it. Hold down the Shift key and click in the brightest purple square in the very bottom row seven squares from the right (as shown in Figure 11.9). This selects all the colors from the first click to the second click.

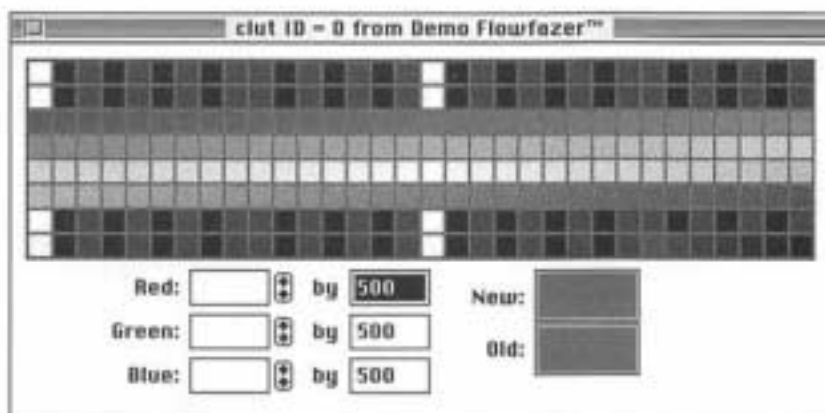


Figure 11.8

The colors in Demo Flowfazer come from this clut resource.

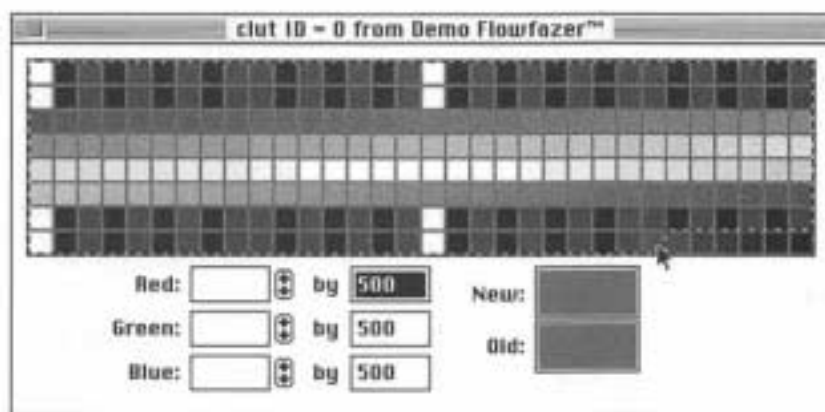


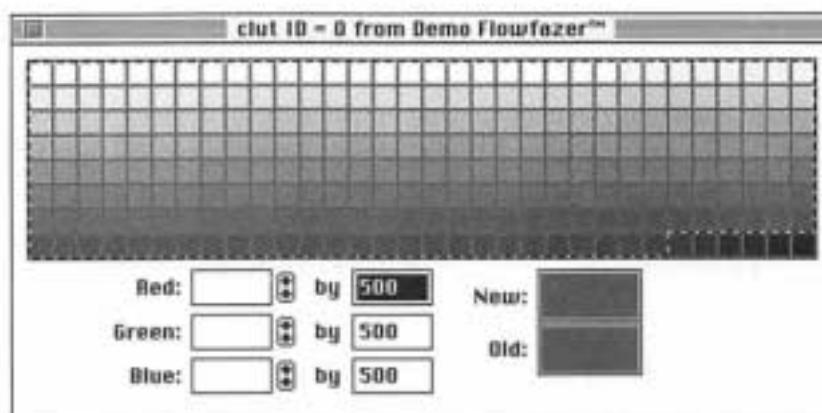
Figure 11.9

Modifying a bunch of colors at once is easy!

Once you've selected all these squares, choose Blend from the clut menu. ResEdit looks at the first and last colors in the selected range and steps all the colors gradually between them to create a smooth blend (see Figure 11.10).

Figure 11.10

The Blend command creates a nice gradient between the colors at extreme ends of the selection.



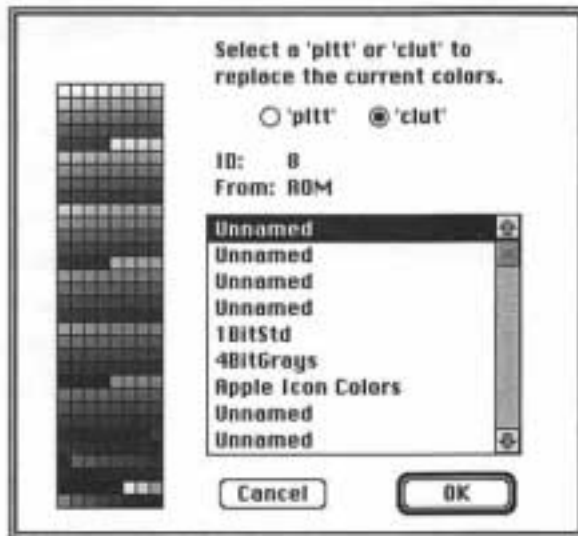
Now let's take a look at what this change looks like when you run Demo Flowfazer. This should give you an idea of how cluts can work to animate the colors of an image. Try making your own patterns and blends and see if you can come up with anything really cool.

The Buddha's Revelations

For a really neat Flowfazer effect, try alternating two colors (such as black and white) one after the other.

Before we move on from the clut resource, let's try one more edit. Open Demo Flowfazer once again and open clut resource 0. Go to the clut menu and choose Load Colors to bring up a dialog box. ResEdit looks at all the clut and pltt resources in the Mac's ROMs, the System, and in ResEdit, and displays a list of them (see Figure 11.11).

With the first item selected, click OK. This replaces everything in the open clut resource with the palette of colors displayed on the left. This happens to be the standard Apple 8-bit System palette located in your Mac's ROMs. Now close Demo Flowfazer once again, saving your changes, and double-click it to see what this color table is going to look like. We think you'll find it a bit much, but it makes the point well.

**Figure 11.11**

All the various cluts in the ROMs, System, and ResEdit.

What's a pltt Resource?

In addition to the clut resource, there is also the pltt resource, which behaves exactly the same way as the clut resource with a couple of minor exceptions. pltt resources are used by the System to determine what colors to use in interface elements such as windows; they also help the System avoid strange side effects when using nonstandard color palettes.

To get an idea of what pltt resources do, find one in an application (ResEdit itself has a couple) and open it. You'll see that there is an additional menu item in the pltt menu labeled Usage. Select a single color out in the middle of the color table and select Usage to bring up a dialog box. No, we're not going to show you that dialog box here. Its use is way beyond the scope of this book. But checking it out might help you get an idea of the complexities involved in the use of color tables.

So at this point we'll plug the nondemo version of Flowfazer. If you like what you see and are curious about the real thing, you can order Flowfazer from various mail-order companies or contact Utopia Grokware directly. Although this software is a bit old, you should still be able to find it. The full version has the ability to switch between clut resources on the fly and has a nice screen saver-like interface. It also has many other neat options for changing the complexity of its patterns.

Flowfazer is to the eye what music is to the ear. This version runs a single example of Flowfazer in 256-color mode only, although the real thing will also run in black-and-white mode on all Macs.

Utopia Grokware
300 Valley Street
Sausalito, CA 94965
415-331-0714 (voice),
415-331-9361 (fax)

Color Palettes Overview

You've seen that a Mac lives and breathes color palettes every time it tries to make a move on the screen. There was a time when working in 24-bit was unheard of, and life was completely limited to these 256-color wonders. You've seen that you can edit them to create your own color choices for painting icons and such. You've also seen how programs can manipulate clut resources to create animated effects. The color palette editor is one of the more technical and powerful features of ResEdit, and a great way to learn about how your Mac really works.

To edit a clut resource:

- Open the resource into ResEdit's color table editor.
- Double-click a color to change it.
- Choose Insert New Color (or press Command-K) to add up to 256 colors.

To blend colors:

- Click to select the first color in your blend.
- Hold down the Shift key and click the second color to which you want to blend the first color.
- Choose Blend from the clut menu to replace all the selected colors between the first and last with a gradient.

To change the clut resource in Demo Flowfazer:

- Open Demo Flowfazer in ResEdit.
- Open clut resource 0 in its editor.
- Modify clut resource.
- Save and close Demo Flowfazer from ResEdit.
- Run Demo Flowfazer to see new clut pattern.

To load a color table from the System or ROMs:

- Open a clut resource you want to replace the contents of or create a new clut resource.
- Choose Load Colors from the clut menu.
- Select the color table you want to load and click OK.

Safety Tips—What Not to Do!

- Don't edit pltt resources. You can cause all sorts of strange problems by doing so.

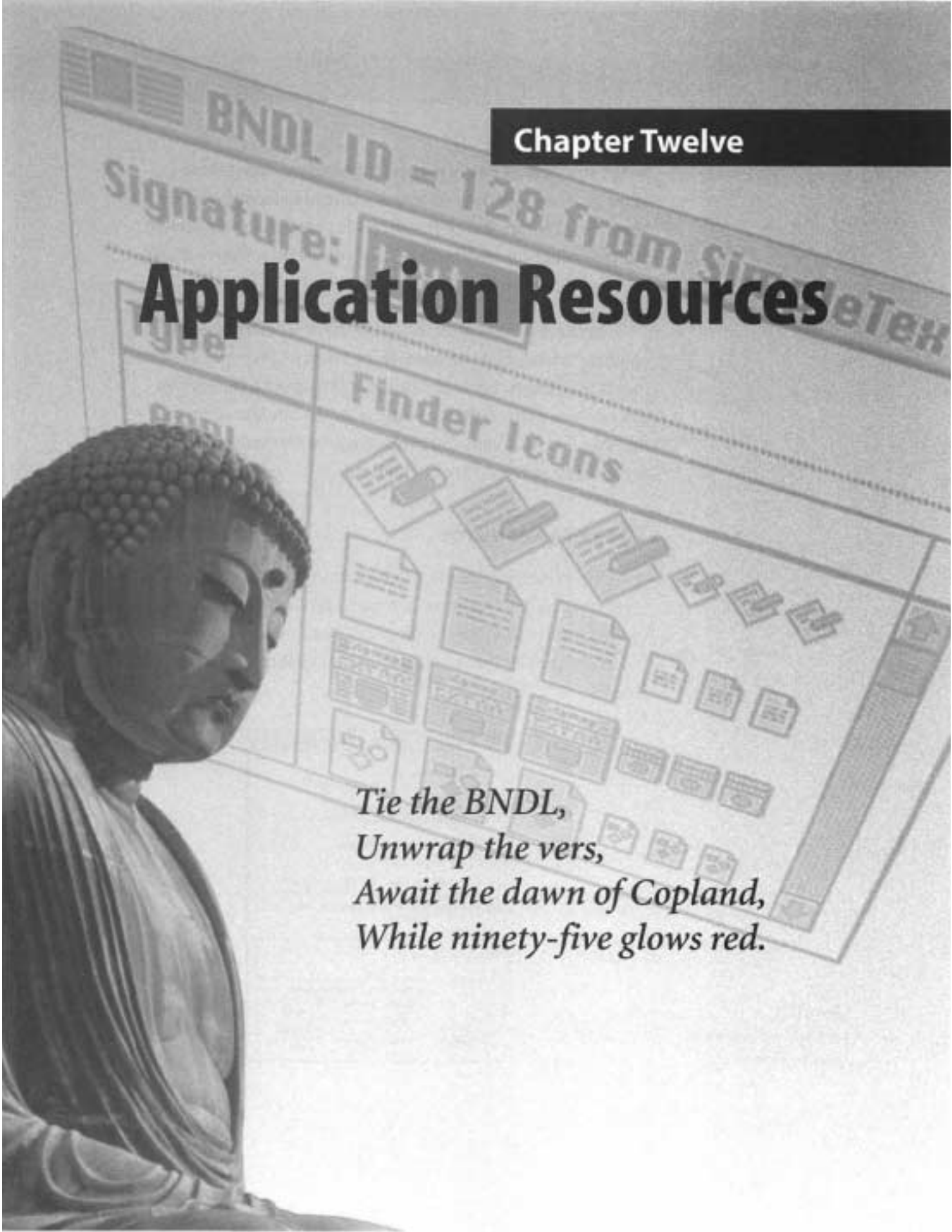
The Buddha's Guide to ResEdit Enlightenment

- Can you find any clut resources in After Dark modules (or other screen saver applications)?
- What is the most mesmerizing clut you can create for Flowfazer?
- Can you make a Flowfazer clut that works with 3-D glasses?
- Can you make Flowfazer run backward?
- What famous rock musician created Flowfazer?
- Can you guess what the pltt Usage options do?
- Can you find a clut resource that controls the way icons darken when selected in the Finder?

Zen and the Art of Resource Editing

Chapter Twelve

Application Resources



*Tie the BNDL,
Unwrap the vers,
Await the dawn of Copland,
While ninety-five glows red.*

So far, you've seen lots of general information about some of the most well-known resources in the Macintosh interface. However, there are lots of resources that are not so well known but can still be interesting to edit, or at least informative about how the Macintosh works.

There are a number of resources that are common to all applications, telling the System things such as the types of documents they create, the icons for those documents, the version information in the Get Info window, or the amount of memory they require. Although you can't do a lot with these resources, the things you can do are often useful.

The vers Resource

One interesting resource that every application contains is the vers resource. This resource contains, obviously enough, the version information for the program. This is what you see when you select the Get Info menu item for an application in the Finder (see Figure 12.1).

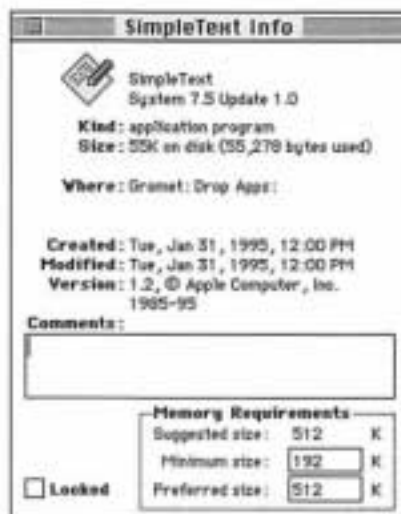


Figure 12.1
The Get Info window in the
Finder for SimpleText 1.2.

To see what the vers resource looks like, open an application (such as the copy of SimpleText you'll find on the CD-ROM) with ResEdit, and open the vers picker. You'll see that there are two vers resources: 1 and 2. Go ahead and open vers 1 (see Figure 12.2).

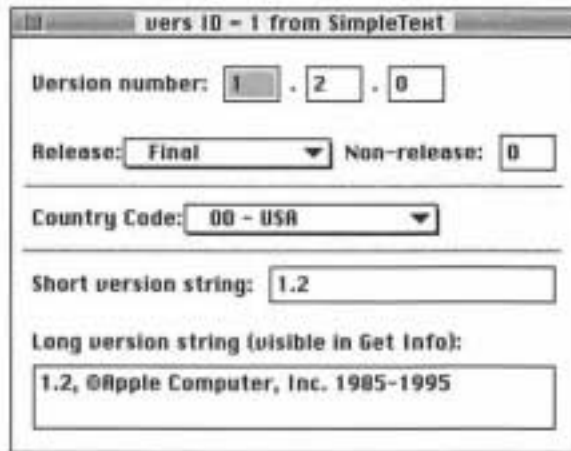


Figure 12.2

vers 1 from SimpleText 1.2, which comes with System 7.5 Update 1.0.

Some applications, including older versions of SimpleText, have only one vers resource (number 1). This just means that they don't take advantage of the information they can provide via vers 2.

The top section of the vers editor contains the three portions of the version number in separate fields. The first field indicates the major version of the software. When this changes, it usually means that the program is completely overhauled. The second field is indicative of the release of the major version, and when it changes, it means that the software is essentially the same but that there have been some features added. The third field is the revision of the particular release, and is usually just for bug fixes without any new features.

The top section of the editor also tells you if the software is final, beta, alpha, or under development. If it's anything other than final, the programmer can include a number in the Non-release

One of SimpleText's file types is *ttro*, which stands for TeachText Read Only. *ttro* documents (the ones with the newspaper icons) can be viewed but not changed.

If you haven't heard the terms before, they're pretty easy. *Development* is the stage in which a lot of the prototyping is done, and when the designers are still trying to figure out everything the program will do. *Alpha* indicates that the basic functionality of the program has solidified, but small features may still be added. *Beta* tells you that the feature set of the program is frozen, and the programmers are looking for bugs. *Final*, of course, means that the program is done (at least as far as the programmers are concerned).

field that indicates the version of the non-release stage (that is, phrases such as "beta 2").

The middle section tells you for which country this version of the program is intended. Although it doesn't mean a lot to change this, it's worth looking at just to see how many countries there are in the list.

The last section contains the short version string, which is used in the Version column in the Finder and isn't required to conform to Apple's Version Numbering Scheme. Thus, you'll often see things like 2.0v2 in the Finder's list view even when that information isn't available in the Get Info window.

Finally, the last portion of the editor contains the long version string. This little blurb is used in the Version section of the Get Info window. This string is the most useful thing you can edit in this resource.

One common use of the long version string is to give credit to the fact that you, the ResEditor, have vastly improved the program with your changes. Another common use of this portion of the resource is to store comments that won't be erased when the desktop is rebuilt. There's a limited amount of text you can put here before the words at the end drop beyond the borders of the field, so keep it short and sweet. The only problem with editing this portion of the resource is that you must then rebuild the desktop for your changes to show up.

vers resource 2, if you have one in your application, tells you version information about where the program comes from. For instance, the version of SimpleText we used for this section says System 7.5 Update 1.0 directly under the name of the program in the information window (refer to Figure 12.1). This string is the long version string of vers 2. You can change this if you want to

give some indication of where a program has come from in the information window.

As you can see, the vers resource can give you a place to store some permanent information about a program. This information is then visible to everyone who looks at the program's information window, which can be very convenient. Best of all, vers resources can be in any file, not just applications, so you can put permanent comments on documents for coworkers to read.

BNDLs and FREFs

Ever wondered how your Mac knows that files created by Microsoft Word have the particular icon that Word icons have? Or how about what dictates what can be dragged and dropped onto an icon? The answers to these questions are buried in the BNDL and FREF, two resources that every program has. A little tweaking with these can produce some interesting results.

Although there are two resources, you need to work directly with only one, the BNDL resource. As with the TEXT and styl resources, the relevant information is combined in the BNDL editor so that you never need to use the FREF editor. Open the BNDL editor for the BNDL resource in the file—probably number 128 (see Figure 12.3).

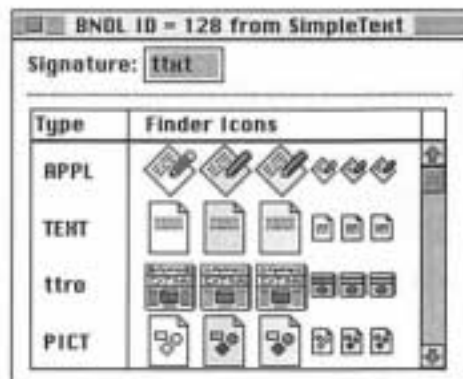


Figure 12.3

SimpleText's BNDL resource, showing various file types associated with the program.

The top of the window contains the signature, or creator type, of the program at which you're looking. For SimpleText, this is ttxt, from the days when this program was named TeachText.

If you see a generic icon for a document or application, it means that the BNDL information has been lost from your Desktop file. To correct this, rebuild your desktop by holding down the Command and Option keys as the Finder starts.

Below this field is a list of all the file types (all given by four-letter codes) that this program can associate with its signature (this is, in fact, a list of FREFs, which is why the easiest way to edit the FREF resources is through the BNDL editor). Every program has at least one entry here, APPL. This is the file type of applications, so the entry for APPL has the program's icon in it. Double-clicking an icon in this editor, by the way, brings up the relevant editor for the icon family.

In addition to the APPL icons, the BNDL contains icons for each of the documents that this program can create. Each one is associated with the relevant file type, so SimpleText includes icons for at least TEXT, PICT, sEXT, and ttro. Your copy of SimpleText may have more depending on the version.

Although SimpleText doesn't list them, some programs (such as ResEdit) have special listings that don't have any icons. If you look at the BNDL in a copy of ResEdit, you'll see that it includes a listing for a **** file type, which doesn't have any icons. In other programs, you might see fold (short for folder) or disk file types with no icons. These mean, respectively, that any files may be dragged onto the application to launch it, folders may be dragged onto it, and disks may be dragged onto it. This is why some programs let you drag files onto their icons, but not folders.

By the way, any file types that are listed in the BNDL can automatically be dragged onto the program, even if it doesn't support the special file types. This is why you can drag any TEXT file, regardless of creator, onto SimpleText's icon to open it.

Bundles of AppleScript

If you work with AppleScript, you may have occasionally wondered if you can set up your drag-and-drop script applications to allow only folders to be dragged onto the application icon. This would eliminate work you might have to do to make sure that the items dragged onto it are indeed folders. With the BNDL resource, it's easy to set this up.

Save your drag-and-drop script as a script application and open that application's BNDL resource. You'll notice that the signature of a drag-and-drop script application is `dplt`. You'll also see that there are entries in the BNDL for `****`, `fold`, and `disk`.

You could, of course, change the entries right in the script application, but this would then affect all programs in which the creator type is `dplt`. In other words, you would cause every droplet on your machine to allow only folders to be dragged onto it. This obviously isn't the right solution!

The way around this problem is to change the signature of your particular script application to something that is not used by any program on your system. For instance, some four-letter shortening of your name or a sequence of numbers (Derrick uses `DERK` and Hans uses `HANS` for these experiments) works nicely.

Once you've done that, delete the `****` and `disk` file types from the BNDL by clicking the row when the cursor changes to a plus sign and pressing the Delete key.

Once you've saved your changes, you'll have to rebuild your desktop to make the changes take effect. This is because the System stores an application's BNDL in the Desktop file for quick access. Once you've done that, you'll find that you can drag folders onto the script application you created, but you won't be able to drag file icons or disk icons onto it, and neither will anyone else.

By the way, Apple actually keeps a registry of creator types. If you're going to do a lot of this type of thing or distribute your scripts widely, you might want to apply to Apple for your own creator type. If your creator type conflicts with the creator of some other program on the desktop, you will cause all sorts of confusion for yourself and your users.

Zen and the Art of Resource Editing

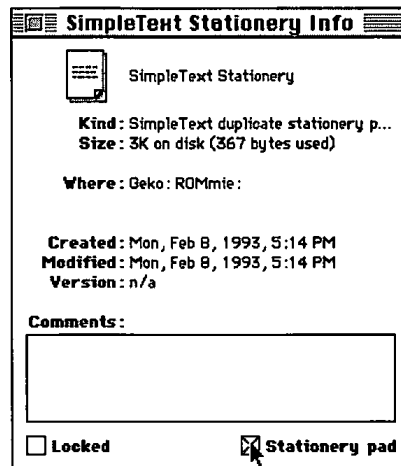
The Script Editor that comes with AppleScript 1.0 completely replaces all the resources in the script application if you edit it and save it out again. Obviously, this trick won't work with Script Editor 1.0. You should upgrade to at least 1.1 to prevent this.

Now your script doesn't need to worry about sorting out whether something is a file or a folder or a disk. If your user drags a combination of folders and files onto this icon, the Finder passes you only the folders.

Bundles of Stationery

Another useful thing to do with the BNDL resource is to give a program an icon for stationery files. These are the files that are created when you Get Info on a document and check the Stationery pad check box (see Figure 12.4). Instead of opening normally when you double-click them, the System makes a copy of the document and prompts you for a new name. You can use this for any boilerplate document you might need.

Figure 12.4
Making a SimpleText document into a stationery pad. Where does that icon come from?



Many applications define their own icons for stationery pads, but some do not, such as Microsoft Word. The System does a good job making an icon for stationery pads if one doesn't exist, but you might want something unique. If the program doesn't define one, what can you do to provide one?

As it turns out, when you turn a document into a stationery pad, the System looks for an icon in the BNDL that has the same

file type, except that the first letter is replaced with a small “s.” If you look in SimpleText’s BNDL resource, you’ll see that it defines a file type named sEXT for stationery pads created from TEXT documents. Similarly, ResEdit defines a file type named ssrc, which is the icon used for stationery pads made from ResEdit documents.

To provide your own stationery pad icon, all you have to do is select Create New File Type from the Resource menu when you are in the BNDL editor. Then click the four question marks to edit the file type information, and change the existing file type to the appropriate one. If you double-click the six squares representing the icons, ResEdit presents a dialog that enables you to choose an existing icon family or create a new one. If you opt to make a new one, ResEdit opens the icon family editor for you.

You can see that the BNDL resource can be moderately useful if you are willing to play with it a little. The one problem with editing this resource is that you need to rebuild your desktop in order for your changes to take effect, but the results can be worth the time.

Application Resources Overview

The System uses a number of resources to get information about applications. BNDL and FREF resources tell the System what icon to show for the program and all its documents, while vers resources give the System the information it needs for the Get Info window.

To add a vers resource to a file:

- Open the file with ResEdit.
- Choose Create New Resource from the Resource menu.
- In the dialog box, type vers and press the OK button.
- When the editor has opened, choose Get Resource Info from the Resource menu.

- Change the ID of the resource to 1 and close the window.
Note: Only vers resources with IDs of 1 and 2 are used by the Finder. A vers resource with a different number will not be used.
- Edit the vers resource appropriately, and close the editor.
(If you want a vers resource that tells where the file comes from, repeat the above steps, but make the new vers resource have an ID of 2.)
- Save your changes and rebuild your desktop.

To add an icon for a file type to the BNDL resource:

- Open a copy of the program you wish to edit.
- Open BNDL 128.
- Choose Create New File Type from the Resource menu.
- Click the ??? string and change it to the file type you wish to add.

To edit the icons for a particular BNDL:

- Open the BNDL editor for BNDL 128 in the application you are editing.
- Double-click the row of icons to the right of a file type within the BNDL.

To add an icon for stationery pads:

- Open the BNDL editor for BNDL 128 in the application you are adding.
- Add a file type for stationery. This is the same as the file type of the document that you are making into stationery, except that the first letter is replaced by a small "s."
- Edit the icons by double-clicking the row of squares next to the new file type and clicking the New button in the dialog box.
- Save your changes and rebuild your desktop.

To rebuild your desktop:

- As you restart your Macintosh, hold down the Command and Option keys until the System asks you if you want to rebuild your desktop.
(When doing this, some of your extensions will probably disable themselves and won't be available again until you restart.)

Safety Tips—What Not to Do!

- Do not distribute modified software. If you change anything in a piece of software, you are legally bound to only use that software yourself.

The Buddha's Guide to ResEdit Enlightenment

- Can you reverse engineer your software by changing the Release pop-up menu back to development in the vers editor?
- Were it not for 1.0, would the program be?
- How many file types does SimpleText (with System 7.5) support and what do they all do?
- Can you edit the contents of a file with the type ttro?
- What does the SIZE resource do for an application?
- Which items in your Control Panels folder are applications?
- Can you find any documents on your Mac that have BNDL resources (and, of course, aren't applications)?
- How does a file or application get a generic icon and how can you fix this?

Chapter Thirteen

Finder Resources

*Magnificent! Magnificent!
No one knows the final word.
The Finder's layout aflame,
Out of the void leap wooden
windows.*

One of the most common targets for resource editing is the Finder. Being part of the System software, it's got lots of interesting resources that you won't find in normal programs. By using ResEdit to edit some of these resources, you can tweak the Finder and make it more useful for your own purposes.

Because most of the resources aren't recognized by ResEdit and come up in the default hex editor, exploring them on your own isn't likely to produce much meaningful information about the way they work. In this chapter, we'll do that part of the job for you. You'll see what some of these otherwise obscure resources can do, and you'll see how to use them to take advantage of some of the Finder's hidden features or personalize it in small ways.

We've cautioned you throughout the book to work only on backups of the programs. No doubt some of you have decided to ignore this warning and work on live copies of things such as the System. In the case of the Finder, you literally can't do this, because ResEdit tells you that the file is already in use. You'll have to make a copy of the Finder and work on that. To put your changes into effect, you'll have to replace the old Finder with the new one and restart.

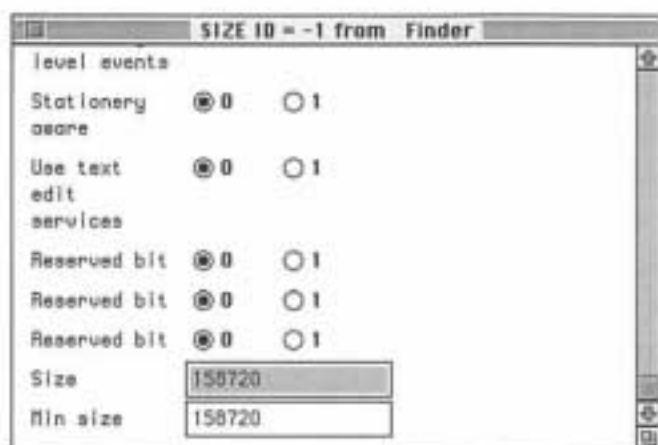
More Memory for the Finder

For most applications on your computer, it's easy to change the amount of memory that the program can request. This is handy when you need to have a lot of windows open in an application, or want it to go faster, or want to send lots of Apple events to it.

The Finder is different. If you use Get Info on the Finder icon, you don't get any fields for changing the amount of memory it uses. So if you want to be able to open more windows in the Finder or just speed it up, you're stuck.

Well, you're really only stuck if you don't have ResEdit handy. The memory each program asks for is determined by the SIZE resource. Because the Finder is basically just another program, it too has a SIZE resource. By editing this, you can give the Finder more memory for its operations.

Open a copy of the Finder with ResEdit (you'll find the Finder within the System Folder), and open the SIZE resource picker. Open SIZE resource -1 and scroll until you're at the bottom of the window (see Figure 13.1).



The two fields at the end of this editor are fairly straightforward. Edit the top one to adjust the size (in bytes) that the Finder grabs if it can. The bottom field indicates how much memory the Finder needs as a minimum. Doubling the top field is probably adequate for most of your needs (at least from our experience), but you might find that even this isn't enough, and you'll have to go through this again. Do keep in mind that every piece of memory you give to the Finder is memory that isn't available for other programs, so you probably don't want to set it too high!

The Buddha's Revelations

If you don't have enough RAM available for an application to start with the suggested RAM allowance, it will use all that it can. If you have less RAM available than its minimum size, it won't start at all.

Figure 13.1

The SIZE resource from the Finder, before you tweak the numbers.

The size of the memory allocated to the program is given in bytes. 1024 bytes is 1K, and 1024K is 1 MB. Thus, the default memory allocation for my version of the Finder is exactly 155K (different versions of the Finder may have different default memory allocations).

Is Your Application “Busy or Missing”?

One of the oldest aspects of the Macintosh interface is that double-clicking a document launches the application program that created that document, often causing it to also open the document that was double-clicked. But what happens when the program that originally created the document is not on the Mac?

Until System 7, you got an error message telling you that “The application is either busy or missing.” Under System 7, all you usually get is a more nicely worded error message: “The application program which created this document is not available.”

However, if the document you double-clicked is a TEXT or PICT file, the System still tells you that the application that created the document is not available, but it then offers you the chance to open the document in SimpleText, which can open files of these types. By clicking OK, you cause the System to launch SimpleText and open the document you double-clicked, just as if it were opening it in the original program.

Of course, lots of programs have the ability to open documents of a certain type, even if the documents were created by other programs. You can edit the Finder so that other kinds of documents are dealt with in the same way. The Finder then asks you if you want to open the document in the application of your choice.

To do this, open a copy of the Finder with ResEdit, and open the fmap resource picker. The fmap that controls this behavior is fmap 17010. Double-click it to open it in the hex editor (see Figure 13.2).

Although fmap 17010 comes up in ResEdit’s none-too-friendly hex editor, you’ll see that it’s actually fairly understandable with just a little effort (for more information about the hex editor

We found out about this a long time ago when a friend of ours was experimenting with a prerelease copy of System 7. However, if you’re exploring on your own, you obviously don’t have this option. Try looking at resources and figuring out how all the pieces might be connected. Because the name of this resource contains “map,” and the contents are file types and the creator code of “txtt,” you might think of where these might be used. Obviously, something that maps file types to SimpleText would fit the bill.

fmap ID = 17010 from Finder					
000000	5445	5854	7474	7874	TEXTttxt
000008	5049	4354	7474	7874	PICTttxt
000010	6C74	7472	6C61	7032	ltrlap2
000018	736A	6F62	7474	7874	sjobttxt
000020	726A	6F62	7474	7874	rjobttxt
000028	716A	6F62	7474	7874	qjobttxt
000030	0000	0000	0000	0000	00000000
000038					
000040					
000048					
000050					
000058					
000060					
000068					

Figure 13.2

fmap 17010 from the Finder controls the Finder's remapping dialog.

and hexadecimal in general, see Appendix C). Each line acts as a map (hence the name of the resource) connecting a particular file type with a particular application signature. When translated, the first line says, “When a TEXT file is opened and the parent application can’t be found, open it with the program whose signature is tttx.” The application that has a signature of tttx is SimpleText, and so its name is in the dialog you see. You can see that PICT files are mapped in the same way.

All you have to do is add a line to this resource to map files of your choice to an application with a particular signature. ResEdit even gives you the means to figure out the file types of the documents and the signature (or creator type) of the applications to which you want to map these orphaned documents.

Let’s look at a simple example: setting Microsoft Word files to open in ClarisWorks. The first task is to find out the file type of Word files, because that’s the first part of the line. Use the Get File/Folder Info menu command in ResEdit’s File menu to help you figure this out.

If you have a document created with Microsoft Word, use the dialog to find it and double-click it (see Figure 13.3). If you don’t have one easily available, you can follow along by looking at the dialogs.

You might have several other entries in this list. These are for other file types related to background processes, and you don’t have to worry about them. They are installed as part of System 7.5 to support QuickDraw GX (even if you don’t install QuickDraw GX).

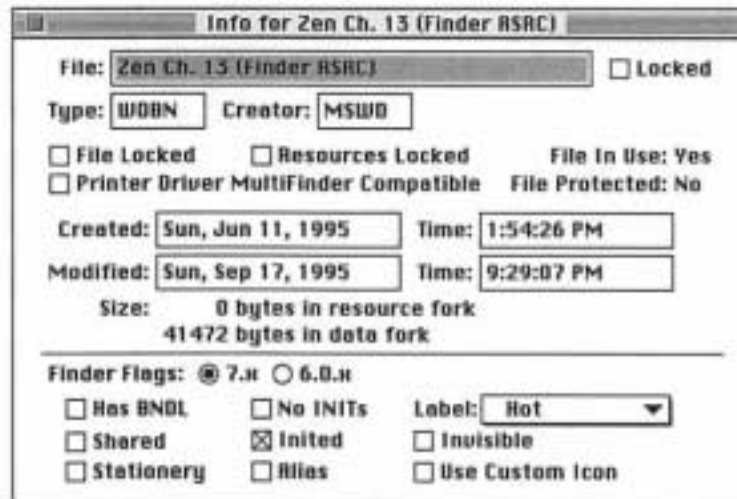


Figure 13.3

ResEdit's Get File/Folder Info window enables you to get the type and creator of any file.

You've seen this dialog before in Chapter 2 when you were setting the type and creator of sound files. At the moment, you're just looking. As you can see in the field labeled Type, Word documents have a file type of WBDN. You can close this window; you will want to use the command again to get the information for ClarisWorks.

Odd creator type, don't you think? Rumor has it that one member of the original ClarisWorks team has a dog named Bobo.

In this case, you're interested in the field labeled Creator, which contains ClarisWorks' creator type, BOBO. Now you have the file type for the documents you want the Finder to recognize, as well as the creator type of the application you want the Finder to open the document in if it can't find Microsoft Word. You're ready to edit the resource.

Back in the hex editor for fmap 17010, place the I-beam cursor just before the last row of zeros by clicking the mouse button when you are at that position; all your entries must be before this last row of 0s. Type WBDNBOBO (this is case-sensitive)—which is the file type you want to open followed by the application signature of the application in which you want the file type to open. The hexadecimal column changes as you type in the

right column. Save your copy of the Finder and replace the one in the System Folder with your new copy. Once you restart, any time you double-click a Microsoft Word document (and assuming you have ClarisWorks on your machine) and the System can't find Microsoft Word, it gives you the option of opening the file in ClarisWorks.

If you're familiar with Macintosh Easy Open, you may wonder why you would need to do this. With Easy Open, opening an orphaned document results in having the System ask you with which application you want to open the document. This is, obviously, far more flexible than the fmap resource. However, Easy Open is an extension, and is therefore disabled when you start up without extensions. If you are using Easy Open, put some of your most common conversions into the fmap resource so that if you have to start without extensions, some of the functionality will still be available.

When Is a Menu Not a Menu?

In Chapter 9, you learned how to work with MENU resources. No doubt you soon went trudging off to edit the Finder's menus to add some Command-keys for things such as Empty Trash and Make Alias (for you pre-System 7.5 types; System 7.5 adds M as a Command-key equivalent). The Finder, however, doesn't use MENU resources, and so you probably came to a screeching halt.

Well, actually, the programmers of the Finder needed more functionality than was encapsulated by the MENU resource, so they created their own resource type called the fmnu. While this doesn't have a pretty editor, you can still do some interesting things with it.

Let's see how you might add a Command-T shortcut to the Empty Trash command in the Finder. Open a copy of the Finder

The Buddha's Revelations

Remember that if the application that did create a document is on your hard drive, double-clicking the document causes that file to be opened with the application that created it.

For techno-weenies, the additional functionality that was needed was to encode Apple event information in the menu command. When you choose a menu item in the Finder, you are sending the Apple event associated with that particular menu command. For a neat hack that uses this functionality to give you a Quit menu item in the Finder, see the next section.

finder ID = 1255 from Finder

000000	0001	0009	0000	04E7	00000000
000008	0000	0000	0753	7065	0000000e
000010	6369	616C	636C	6561	ciaicicic
000018	4006	0000	0843	6C65	0000000e
000020	6166	2055	7000	6560	on UpDen
000028	7074	8006	0000	0C45	ptf0000e
000030	6070	7479	2054	7261	empty Tre
000038	7368	C900	7678	7630	sh...0000
000040	0000	0000	0120	7365	000000-se
000048	6865	1002	4500	0F45	je00000e
000050	6865	6374	2044	6973	ject Dis
000058	6600	7365	7261	1002	kSerru00
000060	0000	0845	7261	7365	00000e
000068	2044	6973	6609	7678	Disk...0x

Figure 13.4
The Finder's Special menu
fmnu resource 1255 in
ResEdit's hex editor.

fnnu ID = 1255 from Finder					
000000	0001	0009	0000	04E7	00000000
000008	0000	0000	0753	70B5	0000000e
000010	5369	616C	636C	6561	c1ac1ac1
000018	4008	0000	0843	6C85	0000001e
000020	616E	2055	7000	0580	on Up0En
000028	7074	8006	0000	0C45	pt00000E
000030	6070	7479	2054	7261	empty Tre
000038	7368	C900	7678	7630	sh...0000
000040	0000	0000	0120	7365	000000-se
000048	6865	1002	4500	0D45	je00000E
000050	6865	6374	2044	6973	ject Dis
000058	6800	7365	7261	1002	k2ser000
000060	0000	0845	7261	7365	00000E
000068	2044	6973	6809	7678	Disk...00

The piece of information that encodes the Command-key equivalent is 3 bytes (a byte is a single character, in this case unreadable by ResEdit and so turned into a generic rectangle) before the name. Change the third byte before the "E" to a "T" (see Figure 13.5).

command should have a big Command-T next to it. Try it! Put something into your trash can (not something you value—perhaps an empty folder) and press Command-T. Your trash should dutifully be emptied. Pretty neat, huh?

fmnu ID - 1255 from Finder					
000000	0001	0009	0000	04E7	00000000
000008	0000	0000	0753	7065	00000spe
000010	6369	615C	636C	6561	clalclea
000018	4006	0000	0843	6C65	#0000Cie
000020	616E	2055	7000	6560	on UpDen
000028	7074	8006	6400	0C45	ptH2E0E
000030	6070	7479	2054	7261	apty Tra
000038	7368	C900	7878	7830	sh_DowD
000040	0000	0000	0120	7365	0000-se
000048	6965	1002	4500	0945	je20E0E
000050	6965	6374	2044	6973	ject Dis
000058	6800	7365	7261	1002	kDseroD
000060	0000	0845	7261	7365	000Eose
000068	2044	6973	68C9	7878	Disk_xc

Figure 13.5

Adding the Command-key for the Empty Trash menu item.

Though fmnu's don't provide the same flexibility as regular MENU resources, you can still do things such as change the text and add Command-keys. Have some fun and experiment to see what you can do.

Quitting the Finder

One of the interesting things you can do with the fmnu resource is to add a functional Quit menu item in the Finder. Why would you want to do this?

One useful benefit of being able to quit the Finder is a quick way to rebuild the desktop. You've no doubt heard that rebuilding your desktop entails holding down Command and Option as you restart your Mac. Although that works, it's not the entire truth. It's holding down the Command and Option key as the Finder starts that actually causes the System to rebuild your desktop.

After you've quit the Finder and no other programs are running, the Finder automatically restarts. If you're holding Command and Option as that happens, you are asked if you want to rebuild your desktop.

The Buddha's Revelations

Quit is the only useful "hidden" menu item you can add to the Finder. All the other Apple events the Finder supports are already covered by menu items.

Being able to quit the Finder gives you another cool advantage. When you use ResEdit to make changes to a copy of the Finder, you can replace the original, and then cause it to quit. As it comes up again, it uses all your changes. That way, you don't have to restart to see the tweaks you've effected in the Finder.

If you are one of those people who likes to work on the original files, being able to quit the Finder will appeal to you, because the Finder restarts only if no other programs are running. Thus, you can start ResEdit, flip back to the Finder, quit it, and use ResEdit to open the original copy of the Finder. Then quit ResEdit and all your changes are available when the Finder restarts!

Back in Chapter 9, we pointed out that you could add menu items all you wanted, but if the programmer didn't write support for the menu item, it won't work. Also, we pointed out that if you switch around any of the items or add separators, forcing items to move down the menu, the command doesn't move with it; rather, it remains in the exact position it was in before. So, you're probably wondering why adding a command in the Finder doesn't give you problems.

The reason this works is that each menu item in an fmnu contains information about which Apple event is sent to the Finder when that menu item is selected. One of the Apple events the Finder supports is the quit event so that you can add a menu item that adds the ability to quit. And because the menu items in an fmnu encode the Apple event and send it to the Finder, you can actually reorder menu items in an fmnu without causing any problems. Just make sure you move all the relevant text!

To add Quit, the first thing to do to take advantage of this nifty feature is to use ResEdit to open fmnu 1252 in a copy of the Finder (Quit, after all, should go in the File menu). As in the previous section, you are presented with the semiunderstandable resource

in a hex editor. Click to the very end of the hex editor, in the middle column (the one with all the hexadecimal numbers), and type the following text: 7175 6974 8100 0000 0451 7569 7400 (see Figure 13.6). Don't put in the spaces, as they appear by themselves. In the rightmost column, you'll see the words quit and Quit, separated by rectangles. The first occurrence is the code for the Apple event, and the second is for the text that will appear in the menu.

Hex	Value	Menu Item
000080	1002 5900 0850 7574	DEVDPut
000088	2041 7761 7900 7878	ReadyBox
0000C0	7830 0000 0000 0120	x000000-
0000C8	5669 6054 8110 4500	findRDF
0000D0	0546 696E 64C9 6669	DFind.f
0000D8	6E6E 8100 4700 0A46	nnR000DF
0000E0	696E 6420 4167 6169	ind Rgal
0000E8	6E00 7878 7830 0000	nR00000
0000F0	0000 0120 7067 7375	000-pgaw
0000F8	0002 0000 0850 6167	4000DFag
000100	6520 5365 7475 70C9	e Setup.
000108	7077 696E 0005 0000	peint.005
000110	0160 7175 6974 8100	Quit 1252
000118	0000 0451 7569 7400	Quit 1252

Figure 13.6

Adding the Quit command to the fmenu resource 1252.

When you've done this, you need to do one other thing before you can save your changes and reboot (that's the last time you'll need to do that to see the changes you've made to the Finder). Go to the top of the hex editor and find the number 11 at the top of the second column of the hex portion of the editor. This tells the System how many menu items are in this menu, and you've just added one. Change this to a 12.

After you've saved your changes and restarted, you now have the ability to quit the Finder on demand. Of course, you may want add a Command-Q shortcut to this menu item. The previous section describes how to add Command-keys to Finder menu items, so you can use your new Quit command for practice.

Clippings

System 7.5 came with a number of extensions, but one of them was particularly useful: the Clippings extension. With this and the

Drag Manager extension running on your machine, you can take selected chunks of text, pictures, or sounds from a Drag Manager-aware application and drag them to the Finder, creating a file that contains just the element you wanted. It's very cool and very intuitive.

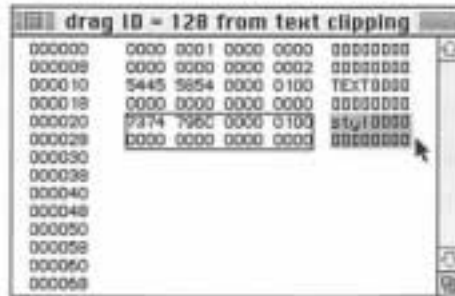
Clipping files can, of course, be opened with ResEdit. There's not a lot to see, but what you can see is useful.

Every clipping file contains a drag resource and a resource that contains the information. Picture clippings have a PICT resource, text clippings have either TEXT or TEXT and styl resources, and sound clippings contain an 'snd ' resource.

If you've made a clipping and want to edit the contents without re-creating the clipping itself, you can edit them with ResEdit. This is really useful only with text clippings, because TEXT and styl resources can be edited directly in ResEdit (as you saw in Chapter 6), while pictures and sounds can't.

Clippings made out of the Notepad do not have styling information.

The drag resource describes what kind of data is in the file. This can be useful if you don't want a text clipping to have any style information. Clippings made by some programs, such as SimpleText, contain all the styling information associated with the document. If you open the drag resource in a text clipping made out of SimpleText, you'll see that it contains the words TEXT and styl. The last two lines tell the System that this clipping contains styled text. If you delete these two lines, the text in the clipping won't have any styling information (see Figure 13.7).

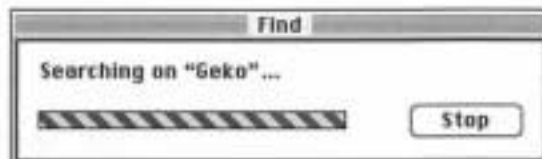
**Figure 13.7**

Selecting the last two lines of the drag resource for styled text—ready to delete.

Changing the Find File Progress Bar

The Finder frequently gets itself involved in a process that is time-consuming, but it can't use the normal progress bar to show you how far it's gotten. This is usually because there is no definite endpoint to the process, such as when you're finding a file on a big hard drive. The next file it finds could be the last file with that name, or it could be the tenth to the last file with that name. There's no way to know in advance how long the process will take.

In those situations, the Finder sports a different progress bar to at least reassure you that it is doing *something* (see Figure 13.8).

**Figure 13.8**

The Finder uses a different progress bar when it's doing a task of indeterminate length.

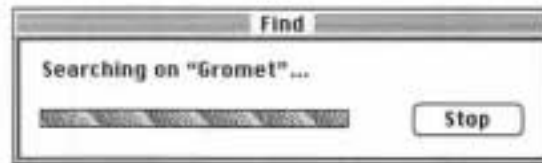
This progress bar is actually similar to the animated cursor it replaces. Instead of endlessly cycling through CURS resources, however, it cycles through four PICT resources. PICT resources 11000, 11001, 11002, and 11003 contain the four frames. Although you can't increase the number of frames in this animation, you can replace the ones that are already there.

In your favorite paint program, create four pictures that are the four frames of the animation (make sure that there's a smooth

transition from the last picture back to the first or your animation will look weird). Select the first one, copy it to the Clipboard, open PICT 11000, and choose Paste. Repeat this with the other three pictures. Save your changes, replace the original Finder, and restart the machine—or quit the Finder and let it restart (see Figure 13.9).

Figure 13.9

Here's the infinite progress bar with our custom PICT resources installed (too bad this book isn't printed in color).



Using the Fonts Folder for More than Fonts

System 7.0 was a big improvement over System 6, but System 7.1, I think, put the finishing touches on what System 7 was intended to be. One of the features that System 7.1 brought to the desktop was the Fonts folder.

No longer does one have to use arcane interfaces or buggy extensions to make fonts available to the System. All you have to do is drag them into the Fonts folder, and they are immediately available (except for programs that are already running).

Because there are many different types of font resources, the System actually just loads all the resources from each font file and makes them available to the System.

This means that if you have a font suitcase with sounds in it, the System simply loads all the sounds and makes them available. In fact, any resource you put into a font suitcase in the Fonts folder is loaded into the System. This provides you with an easy way to add and remove resources from the System without needing to open ResEdit and move resources manually or by working on a copy of the System and rebooting to protect the original.

The only trick is getting hold of a font suitcase. If you've got one on your Mac, you can just make a copy of that and remove all the fonts (by double-clicking it and dragging each of the font files to the trash). If you don't have one, you can use ResEdit to make one (handy, *n'est-ce pas?*), like you did when you were making empty sound files.

To do so, choose New from the File menu in ResEdit. Give your new file a name such as Empty Font Suitcase and click OK. Now you need to change the file type and creator so that the System recognizes this as a font suitcase and not a ResEdit document.

When the window is open, choose Get Info for Empty Font Suitcase (or whatever you named your file) and change the type to ffil and the creator to movr (see Figure 13.10). You can put as many resources as you want into this suitcase, copying them from various sources. Sounds, FKEYs, and even patterns can be copied and pasted into this file with ResEdit. When you're done, close the file. You can then drag this file into the Fonts folder, and all the resources you copied in are now available to the System.

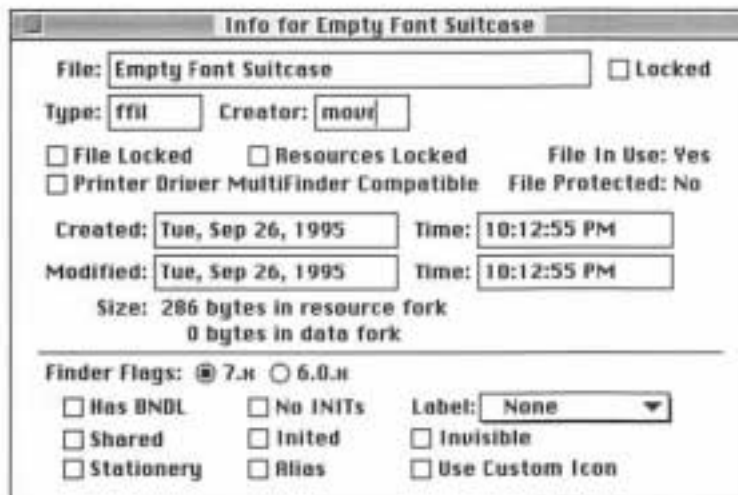


Figure 13.10

To make your new file act like a font suitcase, change the file type to ffil and the creator to movr.

If you put an FKEY resource into the Fonts folder, make sure to include any other resources that the FKEY might need, such as dialogs and sounds.

This handy technique enables you to group resources together and install them into the System without ever touching the System itself. You can combine patterns and sounds, or FKEYs and icons, or anything you need all in one file or in multiple files. Removing them is a bit trickier, because all other programs need to be quit before you can move them out of the Fonts folder, but that's the only drawback to this method.

As an example of how this works, make a copy of your Empty Font Suitcase file. Name it Zen Sounds. Open the Zen Sounds file (in the Book Examples folder on the CD-ROM) with ResEdit. Click the 'snd' resource icon and choose Copy from the Edit menu. If you want, you can open the 'snd' picker to see what sounds we're giving you.

Open the Zen Sounds file (with ResEdit) that you created when you copied the Empty Font Suitcase file. Choose Paste from the Edit menu to put the 'snd' resources into this file. Save your changes and close the file. Drag the file onto your System Folder icon or directly into your Fonts folder.

When you go to your Sound control panel, you'll see two new sounds in your list, Five Blahs and How bad can it be? Those were both in the Zen Sounds file and are now available to your System. You could have added any other resources to this file, and they all would now be available to the System and any other programs on your Mac.

More Interesting Strings in the Finder

You've already seen some of the interesting strings that are kept in the Finder's resource fork. If you haven't gotten around to exploring, here are a couple more pointers to some interesting strings. We found these on a foggy Bay Area day simply by exploring for a little while.

STR# 11250 contains the string “copy” as its 4th item. This is the string that is appended to copies of files and folders made with the Duplicate command. Try changing this to “duplicate” for a custom touch. You can make this empty, but the Finder needs everything to have different names in a single folder. So, if this is an empty string, the Finder makes a file with the same name but with the number 1 appended if you duplicate an item.

Similarly, STR# 20500 contains “alias” as its first item. Remember that if you remove the string entirely, the Finder actually makes files that have the same name as the original, except that they are numbered. To avoid this, it can be changed to something such as a space by itself. This causes the aliases to have the same name as the original, except for a trailing space. This is handy if you don’t want to try to constantly get rid of the “alias” at the end of every alias you make.

One of our favorite Finder strings is the only entry in STR# 14516. It’s the string that is used to provide a sample of a font when you double-click that font in the Finder (see Figure 13.11). Think it’s an odd sentence? Well, we won’t deny that, but it does contain every letter of the alphabet and is more creative than the standard “Quick brown fox jumps over the lazy dog.” Leave it to the Apple folks!

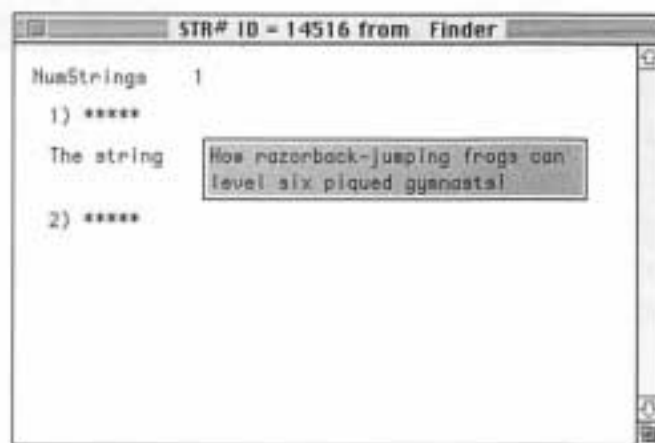


Figure 13.11
One of our favorite STR#
resources in the Finder.

The Buddha's Revelations

If you're using System 7.5.1 and want to disable your ZoomRects, you'll find a utility on the CD-ROM named System 7.5.1 deZoom that lets you do this.

Disabling the ZoomRects

As you may have read in earlier editions of this book, System 6 users have the capacity to disable the animated zoom rectangles that appear every time a window is opened or closed. With the new Finder under System 7, this is no longer easy to get to. Nonetheless, some people have figured out where that information is (unless you're running System 7.5, where it seems to have been obscured again). To disable your ZoomRects in System 7.0 or 7.1 (as they are fondly called), open a copy of the Finder and go to CODE resource number 4. ResEdit tells you it is compressed, but you need to decompress it to make this change (everything has a price, after all). When you open it, you'll be faced with ResEdit's famous hex editor. If you don't know how to use this, you may want to take a second to read Appendix C, "Hex Editing."

After you're all set up, choose Find Offset from the Find menu. Type into the dialog box that you want to find 0078. When ResEdit gets you there, select the following characters (in the hex column): 48E7 1F38. Change these characters to 6000 00E6 by typing over the old characters.

Install your new Finder, restart, and you will be rid of the annoying ZoomRects!

Finder Resources Overview

The Finder, being an integral part of the System software, has all sorts of nifty resources you can play with. Some of the different resource types you've explored in earlier chapters, but the most interesting ones are those unique to the Finder itself. By poking around and experimenting, you can probably find a number of resources that do really cool things when edited properly!

To assign more memory to the Finder:

- Open a copy of the Finder with ResEdit.
- Open the SIZE resource with an ID of -1.
- Scroll to the end of the window.
- Change the value in the Size field to the new memory allocation. (Remember that this number is in bytes, so figure out the number of K you want the Finder to have and multiply by 1,024.)
- Save your changes, replace your old Finder with your new one, and restart the System.

To add to the fmap:

- Open a copy of the Finder with ResEdit.
- Open fmap 17010.
- Before the line of 0s, and at the end of the line above it, type the file type that will be mapped and the creator type of the application you wish the Finder to use in its prompt.
- Save your changes, replace your old Finder with the new one, and restart.

To edit a Command-key in the fmnu:

- Open a copy of the Finder with ResEdit.
- Open up the fmnu you wish to edit. (1251 is the Apple Menu, and 1255 is the Special menu. The ones between these are the menus between them in the menu bar.)
- Find the item to which you wish to add a Command-key.
- Three characters before the first letter of the menu's name, type the letter you want to be the Command-key equivalent for that menu item.
- Save your changes, replace the old Finder with your new one, and restart.

To add a Quit menu item to the Finder:

- Open a copy of the Finder with ResEdit.
- Open fmnv 1252.
- At the very end of the resource, type the following into the middle section of the editor: 7175 6974 8100 0000 0451 7569 7400 (without the spaces; they're added automatically).
- At the top of the resource, find the number 11 in the second column of the middle section and change it to a 12.
- Save your changes, replace the old Finder with your new one, and restart your computer.

To remove the formatting information from a text clipping:

- Open the text clipping with ResEdit.
- Open drag resource 128.
- Select the last two lines in the right section (the last one should say styl followed by four rectangles) and press Delete.
- Save your changes and close the file.

To change the Find File progress bar:

- In your favorite paint program, design four pictures that you want to use as the animation. Make sure they loop well.
- Select the first one and copy it to the Clipboard.
- Open a copy of the Finder with ResEdit and open PICT resource 11000. Choose Paste from the Edit menu to replace the contents of the PICT resource with the first frame of your animation.
- Repeat this for the other frames of the animation, replacing PICTs 11001, 11002, and 11003.
- Save your changes, replace the old Finder with your new one, and restart.

To use the Fonts folder for more than fonts:

- In ResEdit, choose New from the File menu to create a new ResEdit document with the name you choose.
- Select Get Info for [Filename] from the File menu, where [Filename] is the name you gave the file.
- Change the Type field to either ffil or FFIL and the creator to either movr or MOVR, matching the case you used in the Type field.
- Copy and paste resources in from various other sources.
- Save your changes and close the file.
- Switch to the Finder and drag the new file into the Fonts folder in the System Folder (alternately, you can drag it onto the System Folder's icon).

To disable ZoomRects (System 7.0 and System 7.1 only):

- Open a copy of the Finder with ResEdit.
- Open CODE resource 4.
- Select Find Offset from the Find menu.
- Type 0078 into the dialog and press OK.
- Select the following characters in the middle section of the editor: 48E7 1F38.
- Change this to 6000 00E6 (without spaces).
- Save your changes, replace the old Finder with your new one, and restart.

Safety Tips—What Not to Do!

- Don't put a copy of the System and the Finder together in any folder except the System Folder and restart.

The Buddha's Guide to ResEdit Enlightenment

- Can you add a separator line before the Quit menu item?
- Write an AppleScript script that opens the current Finder in ResEdit. (Hint: Buy *The Tao of AppleScript*.)
- If you've got a PowerBook, how would you put a Command-key on the Sleep command? (Hint: Don't look in the fmnu resource; you might want to poke around in the fdmn resource.)
- If you tell the Finder to use 20971520 bytes, how much RAM do you have left (if any) on your machine for other applications?
- If you tell the Finder to use 20971520 bytes and you don't have that much RAM, what will happen?
- Can you set all your orphaned TEXT documents to open in your favorite word processor instead of SimpleText?
- Can you make all your documents from your favorite graphics application open in SimpleText?
- What happens if you double-click a file defined in your fmap resource and the application that created it *is* on your hard drive?
- When is a menu not a menu?
- Can you add Command-key equivalents to the labels in the Label menu?
- What were the names of the guys who programmed the Labels control panel?
- What happens if you set all your label names to be blank?
- What's an Apple event?
- Does the Finder support any other Apple events you can add to your fmnu?
- Can you make your Command-key equivalent be Command-Shift and a letter? How about Command-Option?

- How many different types of clippings are there?
- Can you have more than four frames in your progress bar's animation?
- How do you disable the ZoomRects in System 7.5?

Zen and the Art of Resource Editing

Chapter Fourteen

System Resources

*Moving,
deep in the myst of
seven Systems.*

You've seen how you can make changes to the Finder to effect neat changes on your Mac. While these are fun, they're available only in the Finder. Editing the System (or, rather, a copy of the System) is tons more fun because the changes are global—they impact everything on your Macintosh.

Of course, this means that the changes are potentially more dangerous. Doing the wrong thing could really louse up your applications and cause them to give weird errors. Fortunately, you almost have to deliberately try to wreak havoc on your Mac to really screw something up. As long as you use common sense and don't often let the wolf tempt you from the path on the way to Grandma's house, you'll be fine.

You've already seen a lot of great System customizations throughout the book, but some of the fun things you can do don't fall easily into any of those categories. We'll show you a few more to help round out your ResEdit expertise.

The mcky Resource

Even if you haven't explored the mcky resource, you probably won't be too surprised to discover that it's the resource that provides the options available to you in the Mouse control panel. Leave it to the folks at Apple!

If you open the mcky picker in a copy of the System, you'll see that there are seven mcky resources in the System. If you open the Mouse control panel, you'll discover that there are seven possible settings, ranging from the aptly named Very Slow to Fast. However, if you're one of those Olympic mousers, you may consider Fast to not be as aptly named as Very Slow.

Obviously, each mcky resource corresponds to one of the settings in the Mouse control panel. By editing one of the mcky

resources, you can make your Fast setting act like it should be labeled Supersonic.

To see how to edit this resource, open mcky 6 in its editor (see Figure 14.1). While you've seen other editors of this ilk in ResEdit, they've usually been well labeled so that you can figure out what each field does. The mckys Thresholds aren't as intuitive at first glance.

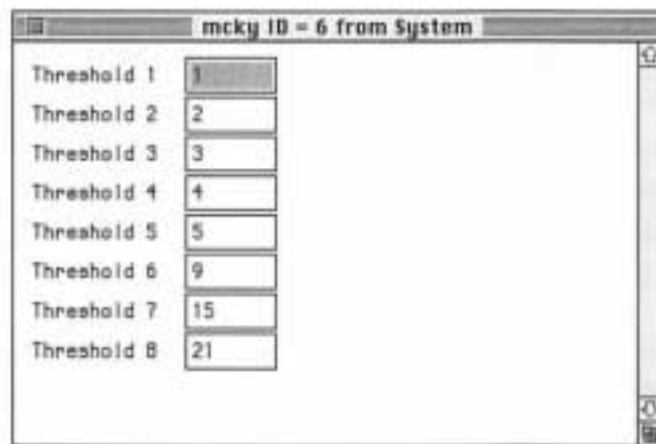


Figure 14.1
mcky 6 before you turbo-charge it.

Unfortunately, we couldn't find a good explanation of what these fields mean. What we did find out with some patience is that lower numbers in the higher thresholds create a faster mouse. We're guessing that these values define the number of pixels the mouse has to traverse before it activates some sort of multiplier that scoots the pointer across the screen.

So, try decreasing each of the last three numbers by half. Save your changes, replace the old System with the new one, and restart. You may need to repeat this process a number of times until you find a mouse speed you like.

Of course, you might not want an ultra-fast mouse. If you want to slow down the fastest setting a little bit (but not as much as

The Buddha's Revelations

If you want your Mouse control panel to have descriptions of your new speeds, you can edit the DITL of the Mouse control panel.

the next lower setting), you can try increasing the final numbers slightly.

Inserting a value of -1 seems to cause that threshold not to be used. Thus the Very Slow setting (mcky 0) has nothing but -1 values in it, implying that a pixel of mouse movement is always a pixel of screen movement.

The Secret About Box

System 7.5 contains a cool little easter egg. Known simply as the Secret About Box, you have to type Secret About Box in some text editor (or the NotePad), select it, and drag it to the desktop to create a text clipping. This brings up the Secret About Box, which is a Breakout-style game featuring the names of the System 7.5 team written on the bricks.

Unfortunately, installing the System 7.5 Update 1.0 caused this trick to be disabled. With a little ResEdit work, however, it can be reinstated into a stand-alone application.

Make a copy of one of your desk accessories, such as the Calculator. Open a copy of the System with ResEdit.

Open timd resource number 12 in the System and select the entire contents of the resource (you can use Select All in the Edit menu) (see Figure 14.2). Copy it to the clipboard and open your duplicated desk accessory.

Figure 14.2

The timd resource, ready to be copied and pasted into your new desk accessory.

timd "System 7.5" ID = 12 from			
000000	0000 0000	0000 0000	00000000
000008	0015 0000	002C 0000	00000, 00
000010	0040 0500	2459 5D54	0F00t lmd
000018	48E7 00C0	48E7 00C0	H0SLH0SL
000020	6100 0032	508F 4C0F	p002P0L0
000028	0300 4E75	48E7 00C0	00HuH0SL
000030	48E7 00C0	6100 0032	H0SL0SL,
000038	508F 4C0F	0300 4E75	P0L000Hu
000040	48E7 00C0	48E7 00C0	H0SLH0SL
000048	6100 0032	508F 4C0F	p002P0L0
000050	0300 4E75	4E56 0000	00HuH0SL
000058	4E56 001C	70E9 4E5E	H/00p0H"
000060	4E75 4E56	0000 70E9	HuH0SLp0
000068	4E5E 4E75	4E56 0000	H"HuH0SL

Open the DRVR picker to find the single DRVR resource. While it's selected, choose Open Using Hex Editor from the Resource menu. Use the Select All command to select the entire contents of the DRVR resource. Paste in the contents of the timd resource.

Save your changes and close the file. Open the new desk accessory, and you should get to see the Secret About Box. Not only that, but you also can play a great game of Breakout. By the way, these steps work on System 7.5 as well.

Changing the Default Creator of Screen Shots

Everyone has, from time to time, found use for the System's built-in screen shot utility. By pressing Command-Shift-3 (thereby activating the FKEY with resource ID 3), you get a PICT file that is the screen exactly as it looked when you pressed the sequence. It doesn't have a bazillion features, but it works.

By default, the System creates the file with a file type of PICT and a creator of ttxt. In other words, it makes a SimpleText document.

This is handy for looking at the picture, but SimpleText probably isn't your favorite graphics program. If you want to do any editing of the screen shot, you need to go through an extra step to open it in your graphics package, either dragging the file icon onto the icon of the program or by launching the program and opening the PICT file from within the application. It's only one extra step, but wouldn't it be nice if the System made a screen shot that was already set to open in your graphics program?

With ResEdit, you can change the FKEY resource in the System so that it creates a file with any creator you want. To do this, open FKEY 3 from a copy of the System file (see Figure 14.3).

The Buddha's Revelations

Although you can change the file creator of the screen capture, you can't change the file type. The actual program in the FKEY encodes images as PICT files.

Figure 14.3

It's this easy to make your screen shots get made with a different creator.

FKEY ID = 3 from System					
000138	820C	FF0C	8885	2050	CD00p n
000140	FF0C	8023	2060	FFFB	0 18 m00
000148	8025	2050	43ED	FFFA	18 PC00™
000150	1200	802E	2060	FFFB	0L 1 m00
000158	8023	30 1F	6600	020A	1800k000
000160	41ED	FF3A	43ED	FFFA	8000C00™
000168	2149	0012	3160	FE94	11001m01
000170	0016	217C	0000	0002	001 0000
000178	0030	8208	0C40	FF00	00p0080-
000180	6700	FF40	4040	6600	g000Jm0
000188	01ED	820C	6600	010A	00p01000
000190	217C	5049	4354	0020	11P1CT0
000198	217C	0474	7874	0024	1100000
0001A0	217C	0000	0002	0030	1100000

Select Find ASCII from the Find menu. Type txt into the first field and click Find Next. ResEdit highlights the found text in the editor.

Change this text to the creator type of your favorite graphics application, making sure to keep the case the same. If you don't know the creator type of the program, you can use ResEdit's Get File/Folder Info dialog in the File menu to figure out what it is.

Save your changes to the System, replace the original with your new copy, and restart. Now when you take a screen shot, the icon on the file is an icon from your graphics program, not SimpleText's PICT file icon. This is your cue that your edit is working.

Welcome to Macintosh

The startup dialog you see when you turn on your Macintosh has become so familiar that most of us ignore it. We notice only if it says something different from we're used to.

After doing all this work on editing your System, the Finder, and all your favorite applications, wouldn't it be nice if this said something other than Welcome to Macintosh? Something that reflected your contribution to the appearance and usefulness of the programs on it?

As it turns out, you can do this via the DSAT resource. This contains the strings that are displayed in that dialog. If you open DSAT resource 0 in a copy of the System and scroll down, you'll notice that the right section of the hex editor contains the strings you're used to seeing: Welcome to Macintosh, Debugger Installed, and Extensions Off (see Figure 14.4). The latter two are seen only when you have MacsBug installed or have held down the Shift key upon startup, respectively.

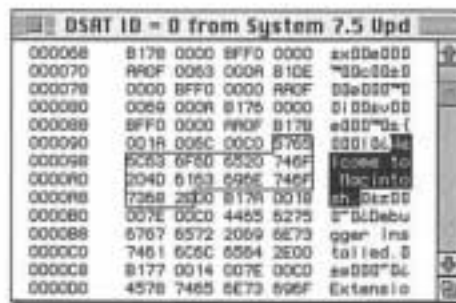


Figure 14.4

DSAT resource 0 in the System 7.5 Update 1.0 file.

Go ahead and change the first string to be more personalized. Ideally, you should keep it the same length or shorter than the original string. You can make it longer, but I noticed when I did that that the color icon was no longer visible in the dialog.

The DSAT resource is used for dialogs that need to be displayed under unusual circumstances, when the Mac might not have full access to the System's resources or display functions. DSAT 2 bears this out; it contains some of the error messages you see, the text from the Force Quit dialog, and the text that is displayed on some machines when the System is shut down ("It is now safe to switch off your Macintosh").

Playing with the DSAT resource can be time-consuming, because you have to restart your Mac every time you want to see the changes. Nonetheless, it adds a final flair to your no-doubt heavily personalized computer.

If you are running System 7.5 with the Update, you need to edit the DSAT resource in the Update, not in the System file.

System Resources Overview

Editing the System is one of the most intriguing things to do with ResEdit, because the changes affect most of the programs that run on your Macintosh. With some judicious prying and pushing, you can really do cool things on your Mac. We've given you a few hints, but you should explore on your own. Do remember, however, that the System is the cornerstone of your Mac: If you destroy it accidentally, you'll need to have a backup handy or spend the afternoon reinstalling System software.

To increase your mouse speed:

- Open a copy of the System.
- Open mcky resource 6 in its editor.
- Decrease the values in the last two or three fields (how much depends on how much faster you want it).
- Save changes, replace your original System with the new version, and reboot.

To make a stand-alone version of the Secret About Box (System 7.5 and greater):

- Make a copy of any of the desk accessories (such as the Calculator).
- Open the copy with ResEdit.
- Open a copy of the System with ResEdit and open timd resource 12.
- Select the entire contents of the resource (using Select All from the Edit menu).
- Choose Copy from the Edit menu.
- Switch back to the desk accessory.
- Open the DRVr resource in the copy of the desk accessory.
- Select the entire contents and choose Paste from the Edit menu.

- Save the changes to the desk accessory and quit ResEdit.
- Change the name of the new desk accessory to Secret About Box and run it.

To change the creator of screen shots made with Command-Shift 3:

- Open a copy of the System in ResEdit, and open FKEY 3 (if you've renumbered this FKEY, open that one instead).
- From the Find menu, choose Find ASCII.
- Type ttxt into the top field and press Return.
- Change the ttxt to the creator of your favorite graphics package.
- Save your copy, quit ResEdit, replace your original System with the new one, and reboot.

To change the text that appears in the startup dialog (System 7.5 with Update 1.0 installed):

- Open a copy of the System 7.5 Update file and open DSAT 0.
- Scroll down until you see Welcome to Macintosh in the far right column (or use the Find ASCII command).
- Change the text to your new message (for best results, make it the same length or shorter).
- Save your copy, quit ResEdit, replace your original System with the new copy, and reboot.

Safety Tips—What Not to Do!

- Don't work on the active copy of the System.

The Buddha's Guide to ResEdit Enlightenment

- Can you change the text in the Mouse control panel to more accurately reflect your new mcky speeds?
- What happens if you change the screen shot FKEY to create a file of a different file type?
- What's your highest score in the Secret About Box?
- What other interesting pieces of text are in the DSAT resources?

Chapter Fifteen

ResEdit Resources

*Buddha's Nirvana
beyond type,
and creator.*

Every Mac application, as you now know, has resources. ResEdit is a Mac application, of course, so the logical conclusion to the syllogism is that ResEdit has resources. Indeed, if you open a copy of ResEdit with ResEdit, you see a wide variety of icons that other applications don't have. ResEdit is designed to be a highly modular program, so its resources are particularly interesting.

By editing some of these resources or moving them in from another file, you can actually add functionality to ResEdit itself.

Installing Editors

ResEdit, as you've seen, is a loosely affiliated cluster of resource editors. Although Apple wrote all the ones that come with ResEdit, it is possible for other people to write their own editors (and even pickers) that you can install into ResEdit.

Let's look at how you might install the cool 'snd' editor that you'll find in the Editors and Templates folder on the CD-ROM. Open the 'snd' editor file from the CD-ROM and dismiss the dialog that warns you that this file is on a locked volume.

You'll see that it contains a number of resources (see Figure 15.1). All of these are vital to the way the 'snd' editor works, so choose Select All from the Edit menu and then choose Copy.

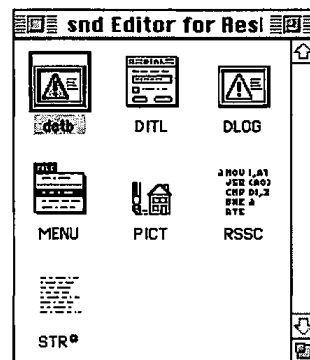


Figure 15.1

The resources in the 'snd' Editor for ResEdit file are all used for this excellent editor.

While you can paste these directly into ResEdit, that's not usually a good idea. If you have to replace your copy of ResEdit, you'll lose all those editors. Also, you're more likely to run into an ID conflict with existing editors, and that could get messy.

A better place to install these editors is the ResEdit Preferences file. Any resources in this file are loaded into ResEdit, in much the same way that any resource in a font suitcase in the Fonts folder is available to the System. If you have to replace or upgrade your copy of ResEdit, it uses the same preferences, and so you'll automatically have all those resource editors available to you.

Because the preferences are loaded at ResEdit's startup, ResEdit Preferences is not being used, and so you can edit it directly without damaging ResEdit. Use the Open command to open the ResEdit Preferences file, which is in the Preferences folder in your System Folder. Choose Paste from the Edit menu. This takes the resources you copied from the 'snd' editor and pastes them into the ResEdit Preferences file, where they'll be available to ResEdit.

Because the contents of this file are loaded when ResEdit launches, you need to save your changes, quit ResEdit, and launch it again. Now, when you open a 'snd' resource, you won't see the familiar hex editor; you'll see the whiz-bang 'snd' editor (see Figure 15.2).

The Buddha's Revelations

As a side note, every ResEdit editor has to have an RSSC resource. This name comes from the two original authors of ResEdit, Roy Sebok and Steve Capps.

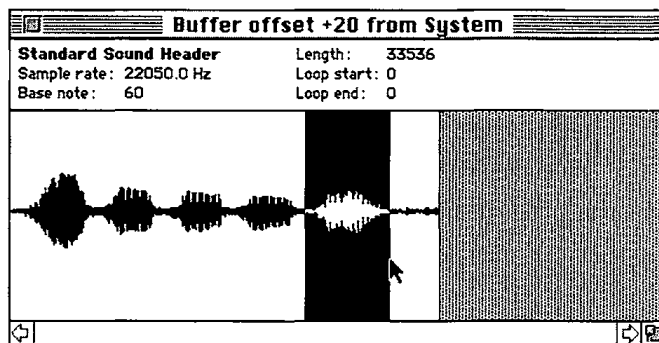


Figure 15.2

Chris Reed's 'snd' editor is a cool addition to any copy of ResEdit.

Zen and the Art of Resource Editing

You can learn more about Chris Reed's 'snd' editor by looking through its documentation on the CD-ROM.

There are a number of public domain editors (and a couple pickers, too) on the CD-ROM that you can plug into ResEdit. In addition to the 'snd' editor, you'll find a great acur editor and picker, a ResEdit editor that enables you to edit the data fork of a file, editors for the various keyboard icons, and a whole slew of other editors and templates for more specialized resources. You can also find editors and templates on various online services. Just follow these same steps and check out some of the great add-ons people have written for ResEdit!

Using Templates

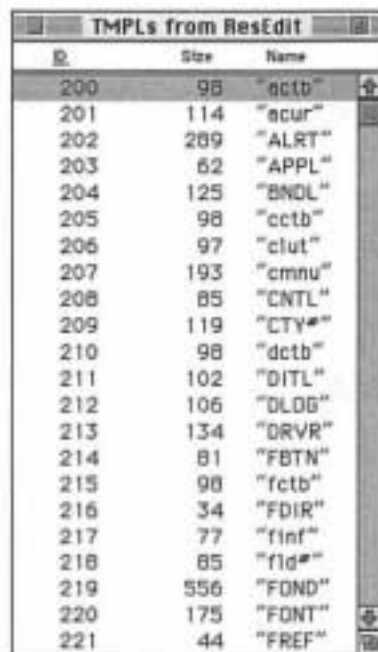
So far, we've referred to everything that enables you to edit a resource as an editor. However, there are actually two types of editors: full-blown editors that feature their own menus and interfaces (such as the ICON editor), and templates that provide a set of fields and occasionally radio buttons that you can use to change the values of the resource. You've actually seen several examples of this type of editor for things such as the acur resource, or the SIZE resource, or any of the other list view editors we've looked at.

Unlike editors, templates consist of a single resource: the TMPL resource. You can design your own templates (in fact, ResEdit uses a template resource to edit TMPL resources), but this requires a lot of technical knowledge and a deep intimacy with the structure of the resource, so isn't usually something people other than programmers do.

If you find templates in the public domain or on electronic services, you install them exactly as you install editors. The one difference is that it doesn't matter if there's a resource ID conflict

with a TMPL resource already in the file, because the important identifier for these resources is the name.

While installing TMPL resources is interesting and can add substantially to what you can easily do with ResEdit, it's a lot more interesting to look for TMPL resources buried in other programs (see Figure 15.3). Programs often have proprietary resources, and sometimes there are TMPL resources in the program that can act as interfaces for those resources. Usually, the programmers have left them in to make resource editing easy for them, but you can often benefit from their efforts.



ID	Size	Name
200	98	"actb"
201	114	"acur"
202	289	"ALRT"
203	62	"APPL"
204	125	"BNOL"
205	98	"cctb"
206	97	"clut"
207	193	"cmnu"
208	85	"CNTL"
209	119	"CTY"
210	98	"dctb"
211	102	"DITL"
212	106	"DLOG"
213	134	"DRVR"
214	81	"FBTN"
215	98	"fctb"
216	34	"FDIR"
217	77	"finf"
218	85	"fld"
219	556	"FOND"
220	175	"FONT"
221	44	"FREF"

The Buddha's Revelations

You can design your own TMPL resources, but the explanation of how to do this is beyond the scope of this book.

Figure 15.3

The TMPL resources included with ResEdit.

If you are exploring a program and see that it has TMPL resources, open the picker. The names of the TMPL resources correspond to the resources that you can look at. For instance, because one of the TMPL resources in ResEdit is named `actb`, you know you can edit `actb` resources (which are in the System) with something other than the hex editor. Open some of the resources

Zen and the Art of Resource Editing

that are covered by TMPL resources, and you may discover some interesting little tidbits.

Notice that you don't have to actually install the TMPL resources from a program into ResEdit Preferences to use it for editing the resources in the program. This is because ResEdit loads the TMPLs it finds in any open documents, so just by looking at the program in ResEdit, you've given ResEdit access to those TMPL resources.

For instance, if you open the Apple Menu Options control panel from System 7.5 with ResEdit, you'll see that it has TMPL resources. If you open the TMPL picker, you'll see that one of the templates is named C&Jp. If you close this picker and look at the resources in Apple Menu Options, you'll see that one of the resource types is C&Jp. Because a template for this resource is available in Apple Menu Options, you can open a C&Jp resource and have a decent editor for it. The C&Jp resource contains most of the same options you'll find in the Apple Menu Options control panel, but it also contains some additional information, such as the names of the menu items for displaying recent applications, documents, and servers. You'll notice that the other resources represented by the TMPL resources in Apple Menu Options are not actually in the file. One of them, FMap, is the Apple Menu Options Preferences, but the DItn resource is leftover from when Apple Menu Options was a commercial product named HAM.

Editing the RMAP

You've seen how modular ResEdit can be with TMPL resources and "external" editors. All you have to do with a TMPL is drop it into ResEdit, and ResEdit uses it when opening the resource. You can use external editors by placing them into the ResEdit Preferences file.

There are lots of other places in ResEdit where simply by dropping in something with the right name, you can get added functionality. One of these is the RMAP. Although you won't often

create an RMAP without some prior information, it's useful to understand how ResEdit does some of the stuff it does for those rare occasions when you will need it.

When you double-click a resource in its picker, ResEdit first looks to see if an editor exists for that resource. If it doesn't find one, it looks at its TMPL resources to see if any have the resource type of the resource you just tried to open as their name. If it finds one, it opens the resource into a template. However, it still doesn't give up if it doesn't find anything in the set of TMPLs.

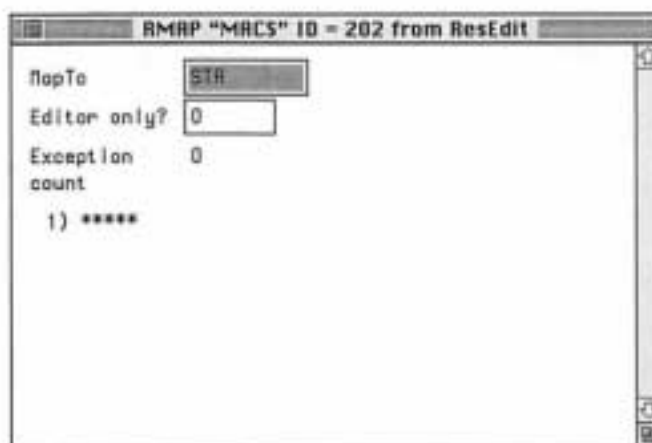
As its final step before resorting to the hex editor, it looks at its list of RMAP resources. If any have a resource name that is the same as the type of resource you are trying to open, it opens the resource in another template or editor.

That's right. RMAPs let you set up mappings so that resources of certain types are opened with some other editor. You'll notice that ResEdit contains, by default, an RMAP with the name MACS, which is a type of resource found in the Finder. If you open it, you'll see that the MACS resource (which is just a string) can be opened in the 'STR' editor. You can try this in a copy of the Finder. Without the appropriate RMAP, the MACS resource would open in ResEdit's default hex editor.

If you open the RMAP resource in ResEdit named MACS, you'll notice that it contains two editable fields (see Figure 15.4). The first is the type of editor in which you would like this resource to open. This particular RMAP tells ResEdit to open MACS resources in the 'STR' editor. The Editor only? field tells ResEdit whether you are opening the resource in an editor or a template. If it's set to 1, you are telling ResEdit to open the resource in another editor.

Figure 15.4

The RMAP named MACS tells ResEdit to open MACS resources into the 'STR' editor. Because the Editor only? field is set to 0, this actually means to open it in the 'STR' template-style editor.



The RMAP resource is also how ResEdit does little tricks such as bringing you into the same icon family editor when you open any of the members of the family. You'll notice that there are RMAP resources for each member of the icon family except icl8. If you look at the editors for each of these, you'll see that they're all mapped to the icl8 editor. Thus, when you double-click an ics8, it opens the icl8 editor (though ResEdit does take care of selecting the ics8 panel of that editor for you).

Of course, this isn't normally the type of thing you'd set up without knowing the structures of the resources pretty well. You'd need to know that some alien resource type is really just a picture before you can tell it to open in the PICT editor and expect good results. As a result, you probably won't be playing with these too much. However, when you need to do it, it's good to know where to look!

Verify Resource

Occasionally, the resource fork of a program or file can become damaged. If ResEdit were to try to open the file, it could crash and burn, potentially damaging its own resources in the process.

Fortunately, ResEdit can provide a preliminary defense against unknowingly opening a corrupted file. When you open a file with ResEdit, you can set the preferences to verify the file as it's opened. This verification process makes sure everything is as it should be in ResEditland, and can even fix problems that it finds (well, most problems).

Usually, you'll witness the verification process only as a progress bar inching its way along. However, if you really want to know what's going on as ResEdit verifies the file, hold down the Option key as you open a file. If you've got the Verify Files when They are Opened preference checked, ResEdit provides you with a dialog that gives you additional information about what it's doing (see Figure 15.5).

If you want to find out the status of a file before you try to open it with ResEdit, you can use the Verify command in the File menu. This will bring up a standard file dialog you can use to select a file to verify. Again, holding down the Option key as you do this shows you the super-cool, secret decoder ring information window.

If you've got the After Dark Star Trek Screen Saver collection, you can easily experiment with this and have some fun with the graphics used in the collection. If you open Color Art in the Shared Resources folder in the After Dark Files folder with ResEdit, you'll see that it contains a type of resource called SiCT. Well, it turns out that these are just PICTs, hidden behind a different resource name to prevent people from editing the graphics. Set up an RMAP to open SiCT resources in the PICT editor, set the Editor only? field to 1, quit ResEdit, restart ResEdit, and open the various SiCT resources from the file. You'll also notice MiCT and PiCT resources scattered about for the masks used in the pictures and B&W Art, respectively. Just don't show your local Paramount representative!

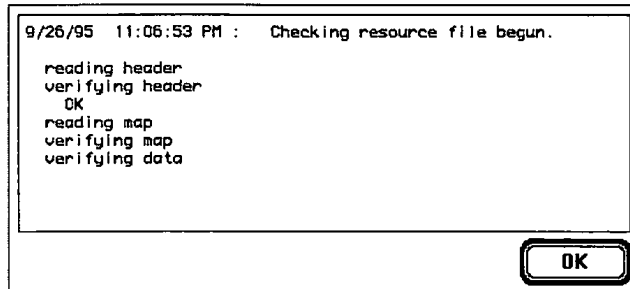
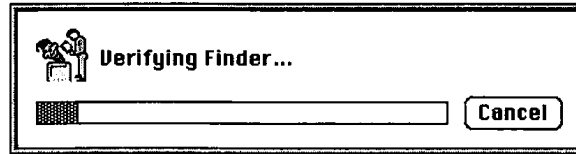


Figure 15.5

Verifying a file is easy, but getting to the optional status screen is a bit trickier.

Setting Up Your Own Icons

You've had plenty of time now to get used to seeing the icons that ResEdit displays in the window you see when you first open a file, but you probably haven't thought much about where ResEdit keeps these icons.

As it turns out, ResEdit uses two types of icons in this window. Black-and-white icons are ICON resources, and color icons are icl4 resources (see Figure 15.6). You'll no doubt remember that when an ICON is used, the color equivalent is the cicon. For some reason, the programmers of ResEdit chose not to adopt that standard.

Whether they're black-and-white or color, you can easily add your own icons to be used for your favorite resources. All you have to do is create a new ICON in a copy of a resource, edit it to your satisfaction, and then set its name to be the same as the resource type to which you want it to be attached. So, if you set up an icon for the fmap resource, you would give it a name of fmap.

**Figure 15.6**

Some of the ICONs in ResEdit are used as the icons that represent resources.

To do the color version of the icon, create an icl4 with the same ID and name. You should do both an ICON and an icl4 so that the icon is visible even if you're running in black-and-white mode. Save your changes to ResEdit, quit, and start your new copy of ResEdit, opening a file that has the resource for which you just created an icon.

As you've seen, ResEdit uses resources quite extensively, far more than most programs. The best part of this is that these resources are usually very easy to edit, enabling you to actually add new functionality to ResEdit. Usually, you won't have much need to edit the resources in ResEdit, but when you do, it's nice to know where things are.

ResEdit Resources Overview

ResEdit, not surprisingly, is chock full of resources. Some of the resources are the normal semicryptic kind unique to any application. However, some are easily editable and can provide

you with the means to edit the actual functionality of ResEdit. You can add editors and templates to ResEdit itself, or customize the way resources are opened in editors, or even provide icons for resource types in ResEdit's windows. Best of all, you don't have to be some sort of super-hacker to do these things. You have to know only how to rename resources and copy and paste.

To install new editors for ResEdit:

- Open the resource file that contains the editor you want to install.
- Choose Select All from the Edit menu and choose Copy.
- Open ResEdit Preferences (which is in your Preferences folder).
- Choose Paste.
- Save your ResEdit Preferences, quit ResEdit, and relaunch it. The new editor is now available.

To look for cool TMPL resources in your programs:

- Open the program in ResEdit.
- If it has an icon for TMPL resources, open the TMPL picker.
- Note the names of the TMPL resources.
- Look for the resource types with those names in the resource window.
- Open a resource of that type and you'll see that the resource can be edited with a template.

To set up an RMAP:

- Open a copy of ResEdit with ResEdit. In the RMAP picker, choose Create New Resource from the Resource menu.
- Choose Get Resource Info from the Resource menu.
- Set the name of the resource to be the same as the type of resource you want opened in another editor.

- Close the Get Resource Info window.
- Set the first field in the template to be the editor to which you want to map this resource.
- Put a 1 in the second field if there is an editor for the resource you will be using to open the other resource type. Put a 0 if there is only a template for editing the resource you will be using.
- Save your changes, quit ResEdit, and start your copy. The RMAP is now available.

To set up an icon for a resource type:

- Open a copy of ResEdit with the original ResEdit. In the ICON picker, choose Create New Resource from the Resource menu.
- Design the icon.
- Close the ICON editor.
- With your new ICON selected, choose Get Resource Info from the Resource menu.
- Set the name of the resource to be the same as the type of resource to which you want to attach this ICON.
- For color icons, make an icl4 via the same process, but match its ID and name to the ICON resource you created (which you should do when you're running in black-and-white).
- Save your changes, quit your current copy of ResEdit, and start your new one.

To verify a file's resource fork:

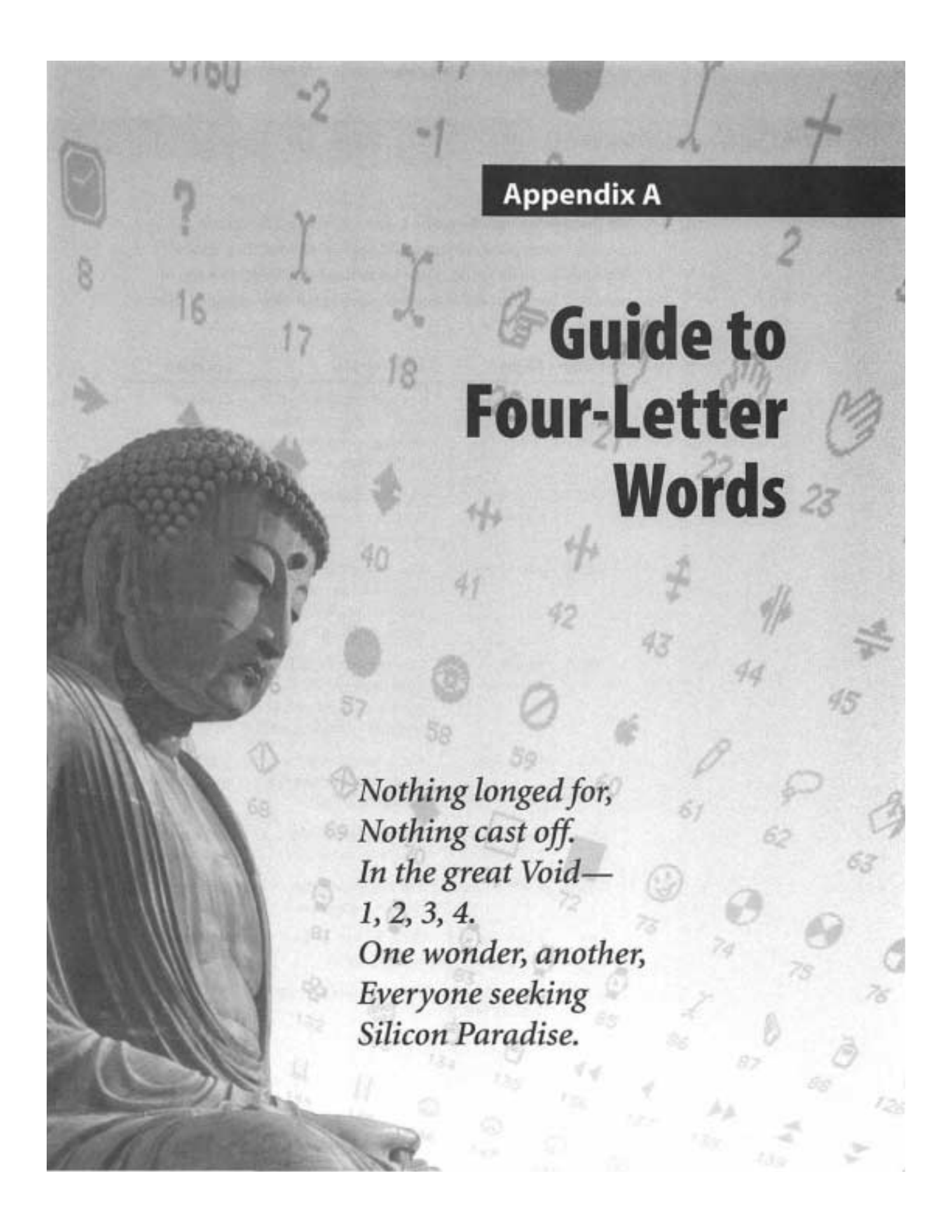
- Choose Verify from the File menu.
 - Select the file you wish to verify.
- or*
- Set up your preferences to automatically verify every file that gets opened.

To see what's happening during a verify:

- When selecting a file to verify (or open, if you have your preferences set to automatically verify), hold down the Option key.

The Buddha's Guide to ResEdit Enlightenment

- Where do you think the name of the RSSC resource comes from?
- What happens if you press Cancel in the middle of doing a verify?
- Can you design your own TMPL resource?
- Who are Roy Sebok and Steve Capps?



Appendix A

Guide to Four-Letter Words

*Nothing longed for,
Nothing cast off.*

*In the great Void—
1, 2, 3, 4.*

*One wonder, another,
Everyone seeking
Silicon Paradise.*

We thought it might be useful to have a list of all the resources covered in this book, so you could easily find what they do and where they can be found. Then we noticed that there are a ton of resources we didn't cover, and we weren't about to—some of those are here, too.

Code	Name	Description	Location
actb	Alert Color Table	The color information used in alerts; double-clicking brings you to ALRT editor	Anywhere
acur	Animated Cursor	The description used to animate a sequence of cursors	System, Finder, applications
ALRT	Alert Info	The information about the positioning of the alert on the screen, similar to DLOG	Anywhere
BNDL	Bundle Resource	Description of which icons to use for the file types associated with a program (including APPL)	Applications
cicn	Color Icon	A color version of the ICON resource; used in dialogs or MENUs	System, applications
clut	Color Look-Up Table	A description of any 256 (or fewer) colors	System, applications
CODE	Code	The actual code that runs an application	Anywhere
crsr	Color Cursors	Color version of the CURS resource	Applications
CURS	Black-and-White Cursors	Description of a pointer	Anywhere

Appendix A Guide to Four-Letter Words

Code	Name	Description	Location
dctb	Dialog Color Table	Color information used by the DLOG resource; opening one takes you to the DLOG editor	Anywhere
DITL	Dialog Item List	A description of the items in a dialog or alert	Anywhere
DLOG	Dialog	The placement and window type of a dialog	Anywhere
DSAT	Dialog Satellite	The strings used in startup and shutdown dialogs, error messages, and force quit dialogs	System
FKEY	Function Key	A small program executed when you type Command, Shift, and the number that corresponds to the resource ID	System
fmap	Finder Mapping	Mapping between files and applications to be used when the creator of a file cannot be found	Finder
fmnu	Finder Menu	The menu resources used by the Finder; they contain information about the Apple events that are sent to the Finder when a particular item is selected	Finder
FOND	Font Description	Description of a font family	System
FONT	Font	A description of the graphics used for each character in a font	System

continues

Code	Name	Description	Location
FREF	File Reference	A single file type/icon combination; the BNDL shows a list of FREFs	Applications
hdlg	Dialog Help	Balloon help for dialog items	Anywhere
hldr	Finder Help	Balloon help for the balloon that is shown when you point to the icon of the program in the Finder	Applications
hmnu	Menu Help	Balloon help for menu items	Anywhere
hrct	Rectangle Help	Balloon help for areas of a screen	Anywhere
hwin	Window Help	Balloon help for windows	Anywhere
icl4	4-Bit Color Finder Icon	The description of the icon to use in the Finder when in 4-bit mode in a large icon view; also the color icons used in ResEdit windows	Anywhere
icl8	8-Bit Color Finder Icon	The description of the icon to use in the Finder when in 8-bit mode in a large icon view	Anywhere
ICN#	Black-and-White Finder Icon	The description of the icon to use in the Finder when in black-and-white in a large icon view	Anywhere

Appendix A **Guide to Four-Letter Words**

Code	Name	Description	Location
ICON	Black-and-White Dialog Icon	The description of icons to use in dialogs or MENUs; used by ResEdit for black-and-white icons associated with resource types	Anywhere
ics#	Black-and-White Small Icon	Description of the icon used when in black-and-white mode and in a small icon view; also used as the icon displayed in the process menu	Anywhere
ics4	4-Bit Small Icon	Description of the icon to use when in 4-bit mode and in a small icon view; also used in the process menu	Anywhere
ics8	8-Bit Small Icon	Description of the icon to use when in 8-bit mode and in a small icon view; also used for the the icon in the process menu	Anywhere
INIT	INIT	Code run as the System starts	System
KBDN	Keyboard Names	The names of the keyboards displayed in ResEdit's keyboard list	ResEdit
KCHR	Keyboard Characters	The mapping of virtual keys to a physical key layout	System
kcs#	Keyboard Icon, Black-and-White	Small black-and-white icon associated with a keyboard layout	System

continues

Code	Name	Description	Location
kcs4	Keyboard Icon, 4-Bit	Small 4-bit icon associated with a keyboard layout	System
kcs8	Keyboard Icon, 8-Bit	Small 8-bit icon associated with a keyboard layout	System
MACS	Mac System	System version number	System, Finder
MBAR	Menu Bar	The MENU resources to be displayed on the menu bar, in order	Anywhere
mcky	Mickey (you know, the mouse)	Information about how fast to move the pointer on-screen relative to the movement of the physical mouse.	System
mctb	Menu Color Table	Colors associated with a MENU resource; double-clicking one opens the MENU editor	Anywhere
NFNT	New Font	A new version of the FONT resource that allows for a wider range of IDs	System
PAT	Pattern	Black-and-white, 8 × 8 pattern (don't forget the space after T!)	System
PAT#	Pattern List	A list of PAT resources used on pattern palettes	ResEdit, System
PICT	Picture	A description of a picture; open while holding down the Option key to see how large the picture is	Anywhere

Appendix A Guide to Four-Letter Words

Code	Name	Description	Location
pltt	Palette	A palette of colors	System
ppat	Pixel Pattern	A color pattern that can be a wide range of sizes, as long as each side is a power of 2	System
ppt#	List of ppats	A list of ppat resources used in areas such as the General Controls control panel in pre-System 7.5 systems	System
RMAP	Resource Mapping	A ResEdit resource that enables you to open an editor for a resource when you double-click some other resource	ResEdit
ROv#	ROM Override	A resource describing a ROM override; fill in the resource type and ID number to use a resource in the System instead of the corresponding resource in the ROMs	System
sfnt	Spline Font	Description of TrueType font	System
SICN	Small Icon	List of small, black-and-white icons	Anywhere
SIZE	Size	The size requirements of the application, as well as some System-required flags	Applications, Finder
snd	Sound	Description of a sound resource (don't forget the space after d!)	Anywhere

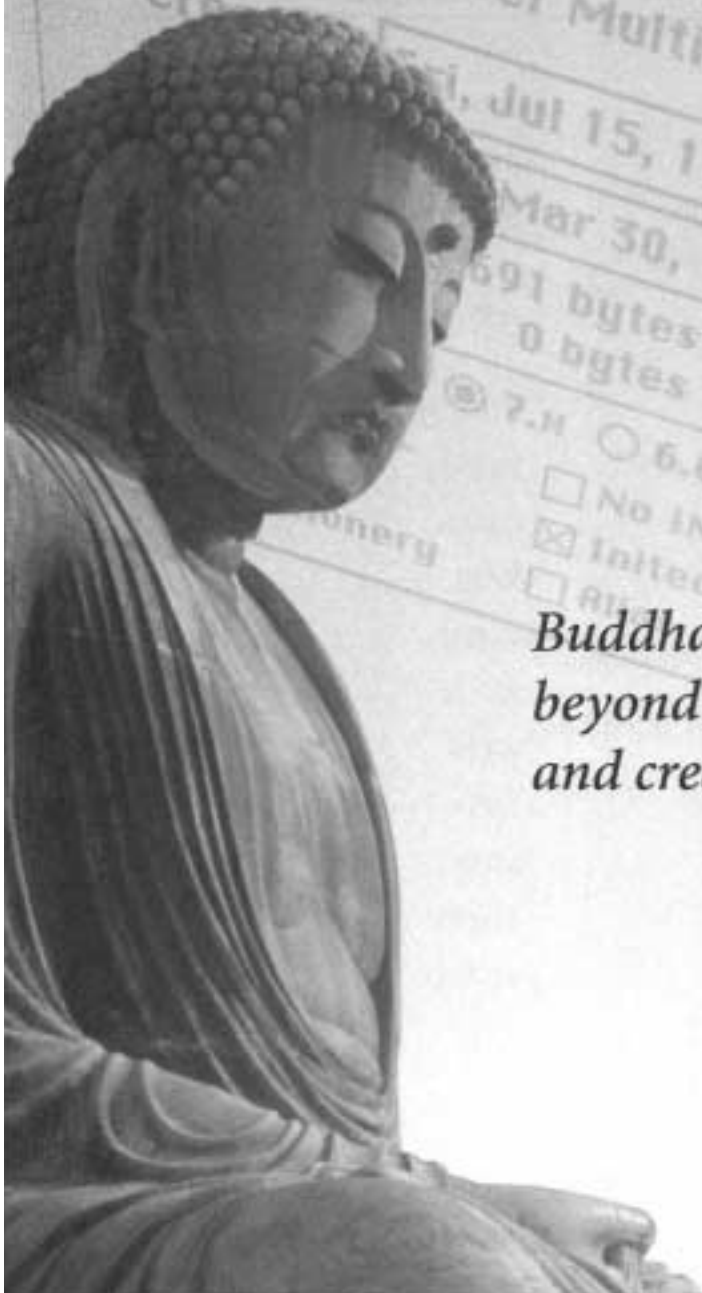
continues

Zen and the Art of Resource Editing

Code	Name	Description	Location
STR	String	A string, or group of characters (don't forget the space after R!)	Anywhere
STR#	String list	A list of STR resources	Anywhere
styl	Style	The style information associated with a TEXT resource; double-clicking one opens the TEXT editor	Anywhere
TEXT	Text	A set of characters; styling information is kept in the styl resource	Anywhere
TMPL	Template	Resource describing a ResEdit template used to edit a resource	ResEdit
vers	Version	Version information for a program	Anywhere

Meet Your Creator

*Buddha's Nirvana,
beyond type,
and creator.*



As you've seen throughout the book, you occasionally need to know creator types when you're hacking your way through the Macintosh jungle. ResEdit gives you the means to find out the creator of your favorite applications via the Get File/Folder Info menu item, but we figured that it would be easier if you had a handy table that listed creator types for some common applications.

Program Name	Creator Code
Acrobat Reader	CARO
Acta	otln
After Dark 3.0	ADr3
AppleScript drag-and-drop script applications	dplt
AppleScript script applications	aplt
BEdit	R*ch
ClarisWorks	BOBO
Eudora	CSOm
Excel	XCEL
FileMaker Pro	FMPR
HyperCard	WILD
Illustrator 5.5	ART5
Illustrator 6	ART6
MacPaint	MPNT
MacWrite Pro	MWPR
Microsoft Word	MSWD
MoviePlayer	TVOD

Appendix B Meet Your Creator

Program Name	Creator Code
Netscape	MOSS
PageMaker 5	ALD5
PageMaker 6	ALD6
Photoshop	8BIM
QuarkXPress 3.3	XPR3
ResEdit	RSED
Script Editor	ToyS
Scriptable Text Editor	quil
SimpleText	ttxt
StuffIt (Deluxe & Lite)	SIT!
StuffIt Expander	SITx

Zen and the Art of Resource Editing

Appendix C

Hex Editing

*The midnight beep,
rarely heard,
came sixteen times today.*

You've seen ResEdit open some complex resources and provide clean and easy-to-use editors for them—editors for icons, menus, dialogs, and even keyboards. You've also seen the somewhat less friendly code editors for those resources that ResEdit doesn't "know" about. Every program has its own custom resources that ResEdit doesn't have an editor for. As a result, ResEdit has a last-resort editor known quite simply as the hex editor. All it does is show you the raw hexadecimal information that a resource contains.

Despite the fact that this editor looks pretty scary when you get into it, it's really not that bad. To see what it's like, open any of the WDEF resources in a copy of your System file. These are the resources that describe the way a window looks, but that won't be evident. What you'll see is something like Figure C.1.

Figure C.1

ResEdit's hex editor sometimes provides meaningful information, but it often appears illegible to mere mortals.

WDEF ID = 0 from System					
000000	600A	0000	5744	4546	"00B4DEF
000008	0000	000F	4E56	FFA4	0000HUI0
000010	48E7	1F7B	429E	0014	H00x8K00
000018	0C6E	0006	000C	5E00	0n000n0
000020	016C	0C6E	0000	000C	010n000
000028	6800	0162	2055	2050	k000 U-P
000030	FF69	266E	000E	49F8	006n0010
000038	06FA	2C2E	0008	3A2E	00 . 00 .
000040	0012	0805	0003	6704	000000g0
000048	4R2B	0071	56EE	FFA5	J+0gU00
000050	0245	0003	302E	0012	0E000 . 00
000058	0240	0007	0C40	0005	00000000
000060	57EE	FFD9	0C7B	3FFF	W0000x70
000068	028E	59EE	FFD8	6308	065000c0

The rightmost column is the only one that might have legible information, though it still probably looks like gibberish. This is the ASCII column, which shows you (between garbage characters) the English equivalents of the hexadecimal code. ResEdit sometimes does a really nice job of this, but other times it looks like nonsense. The middle four columns are the hexadecimal column. It does not contain any information that would be useful to the average user (because most human beings don't think in hexadecimal). The leftmost column shows the offset, in bytes, from the beginning of the resource.

Real Uses of the Hex Editor

For many resources, you won't need to deal with the hex editor, but in the event that you need to, there are a couple things to know. When you're in the hex editor, you get a Find menu. This menu enables you to find by hexadecimal, ASCII, or offset within the resource. That way, you can quickly go to the appropriate area.

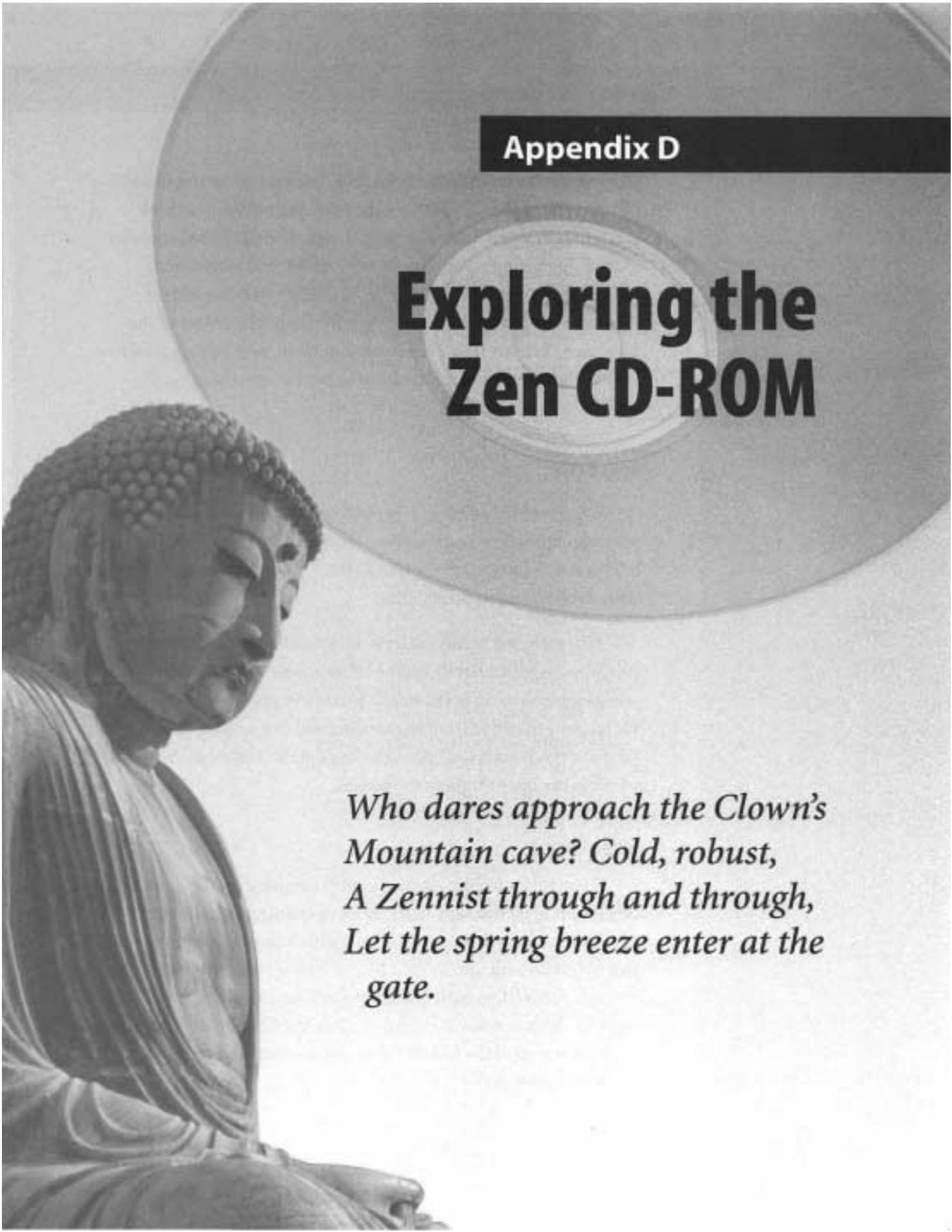
You can also, obviously, edit the text within this resource. However, there are a few little quirks about this. For one thing, when you're typing in either the hexadecimal or ASCII column (you can't type in the offset column), ResEdit translates what you type into the other column. For instance, if you're typing in the ASCII area, the hex column also fills with the hexadecimal equivalent of what you're typing. However, you can actually type into only one area at a time. If you want to copy and paste information somewhere else in a resource, you can paste hex only into the hex column and ASCII into the ASCII column (actually, you can paste hex into the ASCII column, but it's no longer hex when you do).

Typing text has some other differences from what you're used to in a word processor. First, you cannot put a Return into either the hex or ASCII columns. When you hit Return in either column, you are moved to the equivalent location in the other column.

A second change is seen when you're typing into the hex column. Because each ASCII character is represented by two hex characters, when you type something like 5 into the hex editor, ResEdit puts down 05. If you type 5D, it first puts down 05 and then moves the 5 over and tacks on the D, leaving 5D. While this may be distracting, you should keep it in mind if you use the Delete key. The Delete key does not remove one character at a time in the hex column; it removes two characters in one fell swoop. As a result, don't hit the Delete key twice to get rid of 5D.

Zen and the Art of Resource Editing

Beyond that, there's not much more you can do with the hex editor. If you ever wish to open a resource in the hex editor, but it has an editor within ResEdit (which opens first), hold down the Option key while double-clicking the resource in its picker.



Appendix D

Exploring the Zen CD-ROM

*Who dares approach the Clown's
Mountain cave? Cold, robust,
A Zennist through and through,
Let the spring breeze enter at the
gate.*

Okay, so you've enjoyed the book, but the real reason you bought it is because of the CD-ROM in the back, right? Well, we think you'll find that you made the right choice. The CD-ROM contains tons of interesting resources, utilities, hacks, and extensions. Obviously, a complete description of the CD-ROM would be another book, so we'll just show you the highlights. Most of the files have Read Me files associated with them, and we suggest you peruse those for more information about the specific program or file.

ResEdit

Your resource editing career will be short-lived without the program this whole book is about. Fortunately, we've provided the latest version for you on the CD-ROM. Copy it to your hard drive, double-click, and jump right in!

Also included in this folder is an Apple Guide file that will provide you with a handy online reference for some of the stuff you've learned to do in the book. To use this great addition, copy it to the same folder as ResEdit on your hard drive. After that, all you have to do is launch ResEdit—the Zen Guide is listed in the Help menu in the upper right of the screen.

Resources

If you're not feeling creative about designing pretty icons, or don't want to go through all the work of creating an animated cursor of your own, we've included a zillion sample resources for you to browse and use on your Mac. It'll take you a while to look through them all, so we'll quickly point you in the direction of some of the best ones. Be sure to look in the CD Highlights folder on the top level of the CD-ROM for aliases that take you right to the files themselves!

Icons, Icons, and More Icons

You want hip, artistic icons? Have we got a deal for you! In the Icons folder that is within the Resources folder, you'll find oodles and oodles of custom icons. You better plan a long vacation for looking through this set! Our favorites? The Chris Holmes icons (in the H Icons folder) are artistically elegant and clean. Check out the icons he did for BMUG's BBS to see an exhaustive set of folder icons. The Bone Factory icons are odd, bizarre, and sure to shock your friends. There are also some patterns from this guy, so you can make your entire desktop look really strange. We were amazed at the Simpsons icons in the Springfield, U.S.A collection (volumes I–III). Not only is there an icon for practically everyone (and everything) that's ever been on the series, but they're extremely well done.

Mounds of Patterns!

Second only to the number of icons on the CD-ROM is the number of color patterns. The Zen CD comes packed with ppat resources that you can install onto your desktop. Again, the Bone Factory collection provides you with some really strange patterns. We also like the Impressionist Textures for some of the subtleties the artist has come up with. These obviously took a bit of work!

Tons of Cursors! No Waiting!

We've found it amazing what people can do with a 16×16 square in an animation. Included on the CD-ROM you'll find lots of animated cursors that we've scrounged from various sources. The most amazing is the Robin Hood cursor, which tells the story of Robin Hood (sort of the silent, abridged version) in a whopping 271 frames. However, all the cursors are interesting and cool.

And That's Not All!

Icons, cursors, and patterns represent the lion's share of the resources you'll find on the CD-ROM, but there are others lurking about. Keyboards, sounds, and a wide variety of miscellaneous resource collections will make your Mac do things you've never dreamed of. We've also separated out resources that are individual resource artist's collections of System and Finder modifications to help you customize a consistently strange look and feel for your desktop.

Cool Utilities

ResEdit will meet most of your customizing needs, but we've included a number of utilities that provide you with even more opportunity. It's hard to pin down our favorites, but we'll do our best.

For playing with patterns, Desktop Textures was a long-standing favorite until Apple put Desktop Patterns into System 7.5. Desktop Textures is a quick and easy way to get patterns (patterns bigger than 64×64 , even) onto your desktop. Graphics can be pasted in (or copied out), and it's even Apple-scriptable so that you can automate your desktop to look different every time you start up your Mac! Of course, Desktop Textures is best for you pre-System 7.5 folks, as Desktop Patterns offers the same features.

The limitations to using color and animated cursors can be frustrating at times, but two utilities on the CD-ROM ease the pain substantially. CursorAnimator gives you an easy way to set up animated cursors that work globally across your Mac. It's loaded with features, and you'll find it vastly easier to deal with than ResEdit's acur editor. The problem of replacing your arrow cursor is neatly solved by Color Cursor, a little hack that enables you to have a color arrow cursor. This in itself is cool, but the CURS and

crsr resources it uses are easily editable, so you can customize the extension to make your own arrow replacements.

ResEdit spawned a parade of icon and pattern zealots. Zipple, an extension that enables you to replace your Apple icon (the one over the menu) with a color, animated sequence, has its own cult following. Not only will you find Zipple itself on the CD-ROM, but you'll find an amazing and huge collection of Zipple resources to look through.

Menu Bar Patterns is an amusing hack that enables you to put ppat resources in your menu bar. The patterns are stored as a ppt# resource, so you can add your own to customize your menu bar. Although this feature isn't really ResEdit-related, Menu Bar Patterns enables you to make your menu bar transparent. To see how weird this is, you've got to try it!

Although you can't do a lot with it, we've included Aaron, the popular extension that makes most elements of your Mac look like Copland (Apple's forthcoming operating system). Windows are different, the menu bar is colored in, a different default font is used for menu bars, and progress bars are 3-D and puffy. The windows even have a cool ZoomRect effect when you open and close them. You'll be amazed at how ugly your Mac is without it! As an added bonus, Aaron contains a couple TMPL resources so that you can tweak some of the settings that the extension has in its custom resources.

ResEdit Editors and Templates

Having told you how to install editors and templates into ResEdit in Chapter 15, we've also provided you with a number of cool editors for you to play with. Our favorites are the ones in the Chris Reed collection. This guy's got way too much free time on

his hands (something that may have changed now that he's in college). You may also find the kcs#, kcs4, and kcs8 editors interesting if you design your own keyboard layouts.

Authors' Highlights

You can find all the items we've listed here in this appendix and more by going into the CD Highlights folder on the CD-ROM. Here we have put System 7 aliases to all our favorite resources, utilities, and cool stuff. Just double-click and the item opens right up without your having to dig down into the CD-ROM and discover it all for yourself. Enjoy!

Glossary



32-bit QuickDraw — A version of QuickDraw that enables the computer to show more than the 256 colors provided by Apple. See also **QuickDraw**.

Adobe Type Manager — A program from Adobe Systems, Inc., that smoothes out the edges of PostScript fonts you see on your screen. Because QuickDraw printers (ImageWriters, HP DeskWriters, Apple StyleWriters, fax modems, and so on) print what they see on the screen, it will look good in print.

alert box — A message box that appears when the Mac needs to tell you something important. An alert box is similar to a dialog box, except that an alert usually offers only an OK button and no Cancel button or other means of changing the situation. The contents of an alert box are determined by a DITL resource, while its position on the screen is determined by an ALRT resource with the same ID as the DITL resource.

Apple events — Messages that are passed between two programs, enabling one program to communicate with another. Most users don't interact with Apple events, except through a scripting language such as AppleScript or Frontier.

application — A program with which you can perform a task or create a document. Applications can include word processors, spreadsheets, graphics programs, and many more, including ResEdit.

arrow cursor — The cursor used to select and move items on the screen.

ASCII — (American Standard Code for Information Interchange) A system for mapping letters of the Roman alphabet, Arabic numerals, and punctuation. ASCII is understood by

virtually all computers, so it serves as a lowest common denominator for data exchange. The ASCII character set consists of 128 characters.

ATM — See **Adobe Type Manager**.

Balloon Help — A feature of System 7 and greater that provides an informational cartoon-style balloon for objects at which you point. Balloon Help is turned on and off via the Help menu on the right side of the menu bar. Balloon Help can contain normal TEXT resources, styl resources, or PICTs. Applications use hldr, hmnu, hdlg, and hrct resources to control their Balloon Help. Balloons are easily edited with a program named BalloonWriter, available from APDA (Apple Programmers and Developers Association) at (800) 282-2732.

bitmap — An image built out of the pixels on the screen, as opposed to one mathematically constructed.

bitmap font — A font made with bitmap graphics. No matter what size you print a bitmap font in, you are able to see the pixels that make up the characters.

booting up — Starting your computer, either through the on/off switch, the power key, or the Restart menu command (more commonly called “rebooting” because the machine has already booted up).

byte — 8 bits, each of which is a 1 or a 0. An ASCII character takes up a single byte.

check box — An item in a dialog box that enables you to set an option. It looks like a small box that is filled with an X when the option is set and is empty when the option is not set. Check boxes can be found in the palette used by the DITL editor.

clicking — Pressing and releasing the mouse button once.

Clipboard — The System's temporary storage place for text, graphics, and resources. Items go to the Clipboard when you use the Cut or Copy menu commands, and are copied from the Clipboard and put somewhere when you use the Paste command. Although you can use Paste as many times as you want, using Cut or Copy replaces the current contents of the Clipboard and puts the new information onto it. To view the contents of the Clipboard, many applications offer a Show Clipboard menu item in the Edit menu, or you can switch to the Finder and use its Show Clipboard.

close box — A small box in the upper left corner of a window that, when clicked, closes the window.

color dropper tool — A tool in ResEdit's color bitmap editors that enables you to "pick up" a color so that you can use it elsewhere. If you are using the color dropper and click a color in the editor, that color becomes selected in the color palette. If you are using the pencil tool in a color bitmap editor and hold down the Option key, the cursor switches to the color dropper as long as the Option key is held down. The shape of the color dropper is contained in a CURS resource in ResEdit.

Command-key equivalent — A key that you push, along with the Command key, to activate a menu option. These can be edited via the MENU editor for virtually every program, and the fmnu resource for the Finder.

control panel — A utility that controls various aspects of the Macintosh. It can contain something simple such as the speaker volume or something complex such as your screen saver. Some control panels need to load code as the System starts, and are required to be in the Control Panels folder.

Others can be anywhere, because they don't load any code at System startup.

country code — A value that tells the System which country you are localized for (see also **localization**). You can see a list of these in the vers editor.

crosshair — A specific type of cursor that looks like a plus sign. While this is available as a CURS resource in the System, most programs use the copy in the Macintosh ROMs. If you want to edit the crosshair, you need to create a new ROM override via the ROv# resource.

CursorAnimator — A control panel device that you can use to keep a library of animated cursors for use in the Finder.

data fork — The area of a file where all data is stored, such as the text from a word processing document.

dead key — A key that does an action on the key typed after it, such as the key for putting an accent mark over a letter. These are encoded by the KCHR resource and can be edited by typing the dead key while in the KCHR editor or choosing Edit Dead Key from the KCHR menu and selecting the mark you wish to edit.

Desktop file — An invisible file that holds all the icons for the Finder, as well as comments from the Get Info box.

desktop pattern — A pattern used as the backdrop behind files in the Finder. This is contained by ppat 17 in the System file. You can also choose from a list of stored ppats if you use Desktop Patterns under System 7.5 or General Controls in System 7 or 7.1.

dialog box — The normal mechanism by which a user provides information to a program. Dialog boxes can contain text, fields, check boxes, radio buttons, buttons, icons, pictures, pop-up menus, lists, and a wide variety of other items. The contents of a dialog box are kept in a DITL resource, while its position on the screen, coloring, and style are kept in a DLOG resource with the same ID.

Dialog Manager — The part of the Macintosh Toolbox that takes DLOG, DITL, and ALRT resources and displays them properly on the screen, as well as returns information to the program about which items were selected.

document — A file that contains information created by the user through a program. Documents, in order to be edited, need to be opened by some application. Of course, everything is a document from ResEdit's point of view.

dogcow — The logo of Developer Technical Support at Apple Computer. Visible to most through the Option dialog available via the Page Setup dialog, the dogcow's name is Clarus. As you might expect, dogcows say "Moof!"

dpi — Dots per inch; a measure of resolution for printers, monitors, and scanners. This number is the number of pixels that fit in a single inch at that resolution.

dragging — Clicking on an item without releasing the mouse button and moving the mouse around as the button is held down. This has the effect of moving the item in question to some other location.

Dvorak keyboard layout — A specialized keyboard layout said to provide more efficient typing than the standard QWERTY keyboard layout. You can edit your keyboard to act like the Dvorak keyboard layout via the KCHR resource.

editor — Any area in ResEdit that enables you to edit a particular resource. Some editors are very graphical, while others are just lists of options for which you supply values. If no editor is available for a resource, ResEdit opens it in the hex editor, which enables you to look at the raw data in the resource.

eraser tool — A tool available in ResEdit's bitmap editors that erases pixels. Selecting this tool and clicking an area in the editor causes the pixels underneath to become white.

FatBits — A special view of pixels that enlarges them for easy editing. ResEdit's bitmap editors are also called FatBits editors.

file creator — The four-letter code for the application that created a file. This can be viewed with the Get File/Folder Info menu command in the File menu. The file creator of an application itself is also known as its signature.

file type — The four-letter code that represents the type of a file. This can be viewed with the Get File/Folder Info menu command in the File menu. Even programs have a file type, which is APPL for application.

Finder — The component of the System software that enables you to manipulate files, folders, and hard drives. Items can be moved, copied, renamed, erased, and viewed in a number of different ways.

FKEY — A special type of program that acts upon a key sequence, usually Command, Shift, and some number key. The number key you use corresponds to the resource ID of the FKEY resource itself, which is kept in the System file. Obviously, an FKEY with a resource ID of 10 or greater cannot be activated by the normal method.

flag — A special type of variable that can only be true or false. Most flags that the user can change are set via check boxes or radio buttons in dialog boxes.

Font Manager — The portion of the Macintosh Toolbox that controls how fonts are handled, including their display on the screen.

font metrics — The measurements of a character, which include the character width, ascender or upper height of a character, and descender or lower height of a character.

grayed-out — The term for an item that is disabled. The status of an item can be set via the MENU or DITL editor, but is usually done on the fly by the program.

grayscale — A color scheme in which every color is replaced by a shade of gray.

hexadecimal — A numbering system that is based on the number 16 (the one we use normally is based on the number 10).

hierarchical menu — A menu that appears to the side of a particular menu command. Hierarchical menus provide a mechanism for programmers to group related menu items together and tuck them out of the way until needed. The menu is attached to a particular menu item via the MENU editor.

hierarchy (or resource tree) — The way in which the Mac determines where resources are coming from.

hot spot — The active part of a cursor. To set the hot spot, use the hot spot tool, available in the CURS or crsr editors.

Human Interface Design Guidelines — A publication by Apple that specifies guidelines and standards for the interface

of Macintosh programs. Such topics as the menus all programs should contain (Apple, File, and Edit), the proper placement of buttons in dialog boxes, and the key concepts of the Macintosh paradigm are covered by this tome.

I-beam — The cursor that is used for inserting text in a text editor. Although it's visible as CURS 1 in the System, most programs use the one in the ROMs. If you want to edit the I-beam, you need to create an ROv# resource that overrides CURS 1. This forces the System to use the one that you created.

icon — A graphic used to symbolically represent some item on the desktop. Icons can also be used as indications of how important an error message is. The connotations of religious worship don't bear thinking about for those of us who spend all day in front of our computers.

icon families — A group of icons, usually used by the Finder, that associates the black-and-white icons with the appropriate color ones, large and small. The resources in an icon family are the icl8, the icl4, the ICN#, the ics8, the ics4, and the ics#. All the members of an icon family share the same mask.

ID number — The number that uniquely identifies a resource of a particular type in any given file. Every resource has a resource ID, which can usually be seen in the resource picker and can be changed via the Get Resource Info menu item in the Resource menu.

insertion point — The blinking line that shows you where text will appear as you type.

Inside Macintosh — A series of books from Apple that talks about the insides of the Macintosh system.

interface — The way two things communicate. More specifically, the way a user interacts with a particular program.

international codes — The values that tell your Mac where it lives.

K — A unit of measurement for file size. While a picture is worth a thousand words, a K is worth 1,024 bytes, or characters. So would you rather have a picture or a K?

Kanji — Japanese writing style.

keyboard layout — The definition of what characters appear when keys are pressed. This is controlled by KCHR resources in the System file.

localization — The process of setting up System software to work in a different country. At a minimum, this usually entails changing all the text resources in a program to reflect the local language. Other companies change the icons, colors, and even the orientation of elements on the screen to properly reflect the traditions and cultures of the country.

mask — The component of an icon or cursor resource that determines the outline and opacity of the graphic. Most of ResEdit's bitmap editors have a Mask panel that enables you to manually edit the mask of the item.

memory allocation — The amount of RAM an application reserves for itself. An application has a minimum memory allocation, which it must have in order to run, and a preferred memory allocation, which is the amount of memory a program tries to grab if it can. Programs store their memory requirements in the SIZE resource.

menu command — An item in a menu that does some action. These are edited via the MENU editor for most programs and

the fmnu resource in the Finder. However, the action itself is defined by the programmers.

offline — From ResEdit's point of view, the term used when an icon is on the desktop but the disk it is on has been ejected.

picker — A window of ResEdit that lists all the resources of a particular type.

PICT — A picture resource.

pixel — A single dot on the screen (short for picture element).

pointer — Generically, any cursor. Usually, however, it refers only to the arrow cursor.

PostScript — A standard printing language, defined by Adobe Systems, Inc., used by most laser printers.

PostScript font — A font that is based on PostScript language.

push button — An item in a dialog box that either dismisses a dialog box or brings up a new one. When the dialog is dismissed, all the information in it is passed back to the program, letting the program know what the user chose to do. Push buttons are part of the palette available in the DITL editor.

QuickDraw — The routines in the Macintosh Toolbox for drawing both text and graphics on the monitor and on QuickDraw printers.

radio button — An item in a dialog box that enables you to select one of several options. Radio buttons are edited via the DITL editor. In order to be grouped, so that selecting one deselects the others, the radio buttons in the group have to have concurrent dialog item numbers.

RAM — (Random Access Memory) The amount of memory the computer has to work with for calculations, documents, or anything that is done on the fly.

rebooting — The act of restarting.

resource fork — The area of a file that contains all resources. This is what you see when you open a program with ResEdit.

resource tree — A term for how resources are accessed by applications. To find a particular resource, an application first looks in open documents, then in itself, and finally in the System.

ROM — (Read Only Memory) Memory that is fixed within the Macintosh and cannot be changed.

Script Manager — The area of the Mac Toolbox that deals with handling the international codes.

separator line — A gray line that separates different areas of a menu. A menu item can be set to be a separator line via the MENU editor for most programs. The Finder has a special code to define menu items that are separator lines in fmnu resources.

shibboleth — A catchword or slogan used to distinguish members of a particular group.

standard file dialog box — The dialog you use to open or save a file.

suitcase — A special file that contains fonts or desk accessories.

System Folder — A special folder that holds, at a minimum, the System and the Finder. The System Folder also contains special folders for control panels, extensions, startup items, preferences, and fonts. In addition, many programs store their support files in the System Folder so that other programs can easily access them.

System heap — The amount of memory the System reserves for itself.

text view — The view by name, date, kind, or size in the Finder.

Toolbox — The area of the Mac operating system that handles all the elements of the Mac interface.

TrueType — A scalable font technology from Apple and Microsoft built into System 7. Similar to PostScript fonts, TrueType fonts are a geometrical description of the outline of the characters in the font.

typeface — A font.

user interface — See **interface**.

version numbering scheme — Apple's guidelines for assigning a version number to a program or file. The first number of the version number is the major release. The second number is the level of enhancement for the major release. The final number is the level of bug fixes for the enhancement. This number is edited via the vers resource in any program or file.

watch cursor — The cursor that you see while the Macintosh is busy. It looks like a watch.

Zen and the Art of Resource Editing

About BMUG

BMUG is an educational nonprofit organization dedicated to collecting, evaluating, and disseminating information about graphical interface computers. We give users information to help them use their computers efficiently and painlessly. BMUG is not affiliated with any manufacturer, and represents the interests of thousands of users in the United States and in over 50 countries around the world. We offer local activities in the San Francisco Bay area and services for members everywhere. All our meetings are free and open to the public; our disk library is available for sale to anyone. However, technical assistance, newsletters, and our electronic information services are provided for members only.

Membership

BMUG is financed by membership dues and the sale of our educational products. To remain objective in our evaluations of the computer industry, we do not accept advertising in our publications. The majority of BMUG members renew year after year. Read on to find out what keeps them coming back.

Each one-year BMUG membership includes two issues of our highly acclaimed *BMUG Newsletter*, at least one account on the BMUG BBS, and free access to our technical Helpline.

All our meetings, as well as the software library, CD-ROMs, back issues of the *Newsletter*, and other BMUG publications, are available to nonmembers.

Our goal is to provide you with the widest possible array of useful information. Find out why *MicroTimes* listed BMUG as one of the 100 most important influences in the computer industry and said, "BMUG is what every user group dreams of becoming." Please support us by joining BMUG.

BMUG Membership Packages

One-Year Full Membership

\$65

- Two issues of the *Newsletter* per year
- One account on a BMUG BBS (60 min/day)
- Access to the BMUG Helpline

One-Year Contributing Membership

\$45

As a nonprofit organization we are committed to making our services as accessible as possible. We *encourage* those who are not able to join at the Full Membership level to join as a Contributing Member with the same benefits.

One-Year Business Membership

\$150

- Two copies of each *Newsletter* per year
- Up to four accounts on a BMUG BBS
- Access to the BMUG Helpline for 8 people

Online

Our Boston and Berkeley-based FirstClass electronic BBSs (Bulletin Board Systems) are a great way to introduce yourself to the world of telecommunications. The graphical user interface makes the BMUG BBSs easy to use. You can visit any conference area just by double-clicking an icon.

Most of the publicly distributable software is available online, and is subjected to the same intense scrutiny for viruses as our disk-based software library. Both BBSs have multiline capability

for fast modem access. The BBSs are also accessible via the Internet (TCP/IP).

Newsletter

Twice a year, BMUG publishes a newsletter full of interesting articles on a variety of subjects. Unlike other user group newsletters—and unlike other Macintosh publications—the *BMUG Newsletter* is over 300 pages long and contains no advertising. Our newsletter is the heart of our activities; it serves as a medium for the exchange of information and contains some of the most honest, insightful commentary in the entire computer industry.

Joining BMUG is the only way to get the *BMUG Newsletter*, and we work hard to live up to our members' expectations.

Meetings and SIGs

BMUG holds Main Meetings every Thursday on the U.C. Berkeley campus. We also hold South Bay Meetings every third Monday of the month at Apple Town Hall Auditorium in Cupertino. San Francisco meetings are held the last Monday of the month at the Exploratorium. We also offer various Special Interest Groups (SIGs), which focus on particular topics or issues. SIGs meet in the BMUG office unless otherwise specified. All our meetings are free and open to the public; beginners are always welcome.

Technical Assistance

HELPLINE

BMUG volunteers work to provide technical assistance to our membership. We can answer virtually any Macintosh question, simple or advanced. We specialize in both beginner's questions and hard-to-find technical tips.

EMERGENCY DATA RECOVERY

Broken hard drive, crashed floppy?? If you're a member we'll do our best to recover your data, free. There are no guarantees, but we have many tools at our disposal and years of experience using them. An appointment is required, so call our Helpline first. If you can't make it to us, we'll arrange for you to mail it in to us. You can save hundreds of dollars, and we'll even let you join *after* your hard drive crashes. BMUG does RAM upgrades for members as well.

BMUG CD-ROMs

BMUG produces a wide variety of educational CD-ROMs including our PD-ROM with over 600 megabytes of publicly distributable software, articles and other special material and Roadside Resources—The BMUG Internet Collection, which provides you with all the information you'll need to get up and running on the Internet. Many other titles are also available.

Software Library

BMUG has one of the largest collections of freeware and shareware software in the world. The library contains over four hundred 800K disks organized into categories such as Utilities, Fonts, Games, Education, and so on. Disks are available to both members and nonmembers at \$4 each, to cover our costs. Specially priced disk packs are available. BMUG carefully and painstakingly checks everything we sell for viruses.

How to Contact BMUG

Mail: BMUG, 1442A Walnut Street, #62, Berkeley, CA
94709-1496

Fax: (510) 849-9026

Call: (510) 549-2684

(USA Only): (800) 776-2684

How to Reach BMUG BBSs:

Planet BMUG (Berkeley)

Dial Up Access: (510) 849-2684

TCP/IP Address: 206.80.36.0 (TCP Port Number:3004)

BMUG Boston

Dial Up Access: (617) 275-3062

TCP/IP Address: 198.69.254.236 (TCP Port Number:3004)

Zen and the Art of Resource Editing



Index



Zen and the Art of Resource Editing

Symbols

? (question mark), flashing, 8-9
32-bit QuickDraw, 272

A

Aaron extension, 269
actb (Alert Color Table)
 resource, 171, 250
acur (Animated Cursor)
 resource, 89-91, 250
Adobe Type Manager (ATM),
 272
alert boxes, 167-171, 272
aliases, customizing names, 217
Align to Grid command
 (DITL menu), 165
ALRT (Alert Info) resource,
 167-171, 250
animated cursors, 89-91
animation by cycling colors,
 181-184
Apple events, 272
Apple Icon Colors command
 (Color menu), 59
AppleScript, dragging only
 folders onto icon, 195-196
applications, 272
 creator types, 258-259
 Demo Flowfazer, 181-184,
 186
 Desktop Patterns, 76-78
 icons
 displaying, 193-197
 dragging items onto,
 193-197
 editing, 198

 safety guidelines for
 modifying, 199
 version information, 190-193

arrow cursor, 272

ASCII (American Standard Code
 for Information Interchange),
 272-273

ATM (Adobe Type Manager),
 272

B

backing up System file, 3
Balloon Help, 273
Balloon Help command
 (DITL menu), 166
bitmap fonts, 121-122, 273
bitmaps, 273
black-and-white desktop
 patterns, 72-73, 78-79
Blend command (clut menu),
 184
blending colors, 186
BNDL (Bundle Resource)
 resource, 193-198, 250
booting up, 273
Breakout game, Secret About
 Box, 228-229, 232-233
bytes, 273

C

“camera click” sound, adding to
 Sound control panel, 26-30
CD Highlights folder
 (on CD-ROM), 270

- character grid, finding keys, 140
- check boxes, 159, 273
- Choose Icon command
 - (MENU menu), 151
- cicn (Color Icon) resource, 53-56, 250
- cicn menu commands
 - Delete B & W Icon, 58
 - Icon Size, 58
- ClickChange utility
 - (Dubl-Click Software), 92
- clicking, 274
- clipboard, 274
- Clippings extension, 211-212, 220
- close box, 274
- closing windows, disabling zoom
 - rectangles, 218, 221
- clut (Color Look-Up Table)
 - resource, 178-187, 250
- clut menu commands
 - Blend, 184
 - Load Colors, 184
- CODE resource, 250
- Color Cursor utility, 88-89, 268-269
- color dropper tool, 274
- color look-up tables (clut), 176-177
 - loading from System or ROMs, 187
- Color menu commands, 58-60
- color palettes
 - animation by cycling colors, 181-184
 - editing, 178-181
 - System palette, 177
- Color Picker command
 - (Color menu), 59
- colors
 - blending, 186
 - cursors, 88-89
 - dialog boxes, 171, 173
 - dialog icons, 53-56
 - menus, 147, 150
- Command-key equivalents, 148-149, 274
 - editing, 219
- commands, 280-281
 - cicn menu
 - Delete B & W Icon, 58
 - Icon Size, 58
 - clut menu
 - Blend, 184
 - Load Colors, 184
 - Color menu, 58-60
 - CURS menu, Try Pointer, 88
 - DITL menu
 - Align to Grid, 165
 - Balloon Help, 166
 - Grid Settings, 165
 - Renumber Items, 164
 - Select Item, 165
 - Show All Items, 165
 - Show Item Numbers, 165
 - View As, 165-166
 - Edit menu
 - Copy, 32, 34
 - Duplicate, 28
 - Paste, 32
 - Select All, 54
 - File menu
 - Duplicate, 26
 - Get info for [file name], 35-36
 - New, 34
 - Save, 36

Zen and the Art of Resource Editing

- grayed-out, 278
- KCHR menu
 - Convert to Dead Key, 136
 - Edit Dead Key, 135
 - Remove Dead Key, 136
 - Remove Unused Tables, 132-133
 - Uncouple Modifier Keys, 133
 - View As, 133-134
- MENU menu, 151-152
- PAT menu, Try Pattern, 73
- ppat menu
 - Pattern Size, 75
 - Try Pattern, 75
- Resource menu
 - Create New Resource, 54, 178
 - Get Resource Info, 28, 40-41
 - Insert New Color, 180
 - Insert New Field(s), 91
 - Insert New Pattern, 79
- snd menu, Try Sound, 27
- control panels, 274-275
 - CursorAnimator, 92, 268-269, 275
 - General Controls, pattern lists, 78-80
 - Keyboard, 126
 - Mouse, 226-228
 - Sound, adding “camera click” sound to, 26-30
- controls, 160
- Convert to Dead Key command (KCHR menu), 136
- copies, customizing names, 217
- Copy command (Edit menu), 32, 34
- copying
 - icon resource contents, 67
 - System file for editing, 26-27, 42-43
- country codes, 275
- Create New Resource command (Resource menu), 54, 178
- creator types, 32-33, 258-259, 277
 - changing, 44
 - RSED, 35
- crosshair, 275
- crsr (Color Cursors) resource, 88-89, 250
- CURS (Black-and-White Cursors) resource, 84-88, 250
- CURS menu, Try Pointer command, 88
- CursorAnimator control panel utility, 92, 268-269, 275
- cursors, 84
 - animated, 89-91
 - arrow, 272
 - colors, 88-89
 - crosshair, 275
 - editing, 91-94
 - hot spots, 85-87, 94, 278
 - I-beam, 279
 - masks, 87-88, 94
 - on CD-ROM, 267
 - pointers, 281
 - watch, 283
 - editing, 93-94
- custom icons, 64-65, 68

D

- data forks, 275
- dctb (Dialog Color Table)
 - resource, 171, 251
- dead keys, 134-136, 140, 275
- default screen-shot creators,
 - changing, 229-230, 233
- Delete B & W Icon command (cicn menu), 58
- deleting formatting from text
 - clippings, 220
- Demo Flowfazer application, 181-184, 186
- desktop
 - patterns, 275
 - black-and-white, 72-73, 78-79
 - color, 74-80
 - rebuilding, 64, 67-68, 199
- Desktop file, 275
- Desktop Patterns application, 76-78
- Desktop Textures utility, 268
- dialog boxes, 276
 - colors, 171
 - components, 158-160
 - check boxes, 273
 - editing, 160-167, 172-173
 - numbering, 169-170
 - pushbuttons, 281
 - radio buttons, 281
 - standard file, 282
 - types, determining, 167-169
 - Welcome to Macintosh,
 - changing text, 230-231, 233
- dialog icons
 - color, 53-56
 - editing, 48-52
 - masks, 56-60, 67
- Dialog Manager, 276
- DITL (Dialog Item List)
 - resource, 158-167, 251
- DITL menu commands
 - Align to Grid, 165
 - Balloon Help, 166
 - Grid Settings, 165
 - Renumber Items, 164
 - Select Item, 165
 - Show All Items, 165
 - Show Item Numbers, 165
 - View As, 165-166
- DLOG (Dialog) resource, 167-171, 251
- documents, 276
- dogcow, 276
- dpi (dots per inch), 276
- Drag Manager extension, 212
- drag resource, 212
- dragging, 276
 - items onto application icons, 193-197
- DSAT (Dialog Satellite) resource, 231, 251
- Dubl-Click Software's
 - ClickChange utility, 92
- Duplicate command
 - Edit menu, 28
 - File menu, 26
- duplicates
 - customizing names, 217
 - resources, 43
- Dvorak keyboard layout, 276

E

- Edit Dead Key command (KCHR menu), 135
- Edit Menu & MDEF ID command (MENU menu), 152
- Edit menu commands
 - Copy, 32, 34
 - Duplicate, 28
 - Paste, 32
 - Select All, 54
- editable text (dialog boxes), 160, 172
- editing
 - application icons, 198
 - Command-key equivalents, 219
 - cursors, 91-94
 - custom icons, 68
 - dialog-box components, 160-167, 172-173
 - Finder menus, 207-209
 - fonts, 112-119, 122
 - icon families, 63-65
 - icon resources, 67
 - pattern palette, 79
 - patterns, 79-81
 - PICT resources, 37-39
 - System file, 26-27, 42-43
 - watch cursor, 93-94
- editing area of icon editors, 50
- editors, 277
 - acur, 89-90, 92-93
 - BNDL, 193-197
 - cicn, 54-60
 - clut, 178-181
 - crsr, 88-89
 - CURS, 85-88
 - DITL, 160-167
 - DLOG, 167-173
 - FOND, 114-117
 - FONT, 112-114
 - hex, 262-264
 - ICON, 18-21, 49-52
 - icon families, 61-65
 - installing, 236-238, 246
 - KCHR, 128-129
 - mapping resources to, changing, 240-242
 - MENU, 146-152
 - on CD-ROM, 269-270
 - 'PAT', 73
 - ppat, 74-76
 - sfnt, 117-118
 - SICN, 66
 - STR#, 100
 - TEXT and styl, 101-102
 - vers, 191-193
- eraser tool, 50, 277
- extensions
 - Aaron, 269
 - Clippings, 211-212, 220
 - Drag Manager, 212
 - Zipple, 269
- eyedropper tool, 55-56

F

- families of icons, 279
 - Finder icons, 61-65
- FatBits, 277
- FATs (Font Association Tables), 108, 115-117
- File menu commands
 - Duplicate, 26
 - Get info for [file name], 35-36

- New, 34
- Save, 36
- files
 - application version
 - information, 197-198
 - clipping, 211-212, 220
 - creator types,
 - 32-33, 258-259, 277
 - changing, 44
 - RSED, 35
 - Desktop, 275
 - duplicates, customizing
 - names, 217
 - file types, 32-33, 277
 - changing, 44
 - rsrc, 35
 - PostScript fonts, 119-121
 - resource, creating, 44
 - resources in, 22, 30-31
 - sound, creating from 'snd'
 - resources, 33-36
 - stationery, giving applications
 - icons for, 196-198
 - System
 - backing up, 3
 - copying for editing,
 - 26-27, 42-43
 - replacing, 29-30, 43
 - restoring, 8-9
 - verifying, 243, 247-248
 - without originating
 - applications, opening,
 - 204-207
- Find File progress bar, changing,
 - 213-214, 220
- Finder, 277
 - memory, adding,
 - 202-203, 219
 - menus, editing, 207-209
 - quitting, 209-211, 220
 - strings, 216-217
- Finder icons, 60-65
- finding keys on character grid,
 - 140
- FKEY (Function Key) resource,
 - 39-41, 229-230, 251, 277
- flags, 278
- flashing question mark, 8-9
- fmap (Finder Mapping) re-
 - source, 204-207, 219, 251
- fmnu (Finder Menu) resource,
 - 207-211, 219, 251
- folders
 - CD Highlights
 - (on CD-ROM), 270
 - dragging onto AppleScript
 - icon, 195-196
 - duplicates, customizing
 - names, 217
 - Fonts, grouping/installing
 - resources, 214-216, 221
 - System Folder, 283
- FOND (Font Definition) re-
 - source, 107-108, 251
 - adding FONT resources to,
 - 122-123
 - editing, 114-117
 - renaming/renumbering,
 - 118-119
- Font Association Tables (FATs),
 - 108, 115-117
- Font Manager, 278
- FONT resource, 107, 251
 - adding to FOND resources,
 - 122-123
 - editing, 112-114
- font suitcases, creating, 215

Zen and the Art of Resource Editing

fonts, 111-112

- bitmap, 121-122, 273
- displaying samples, 217
- editing, 112-119, 122
- IDs, 107-110
- metrics, 278
- original system, 107
- outline, 110-111
- PostScript, 119-121, 281
- TrueType, 283

Fonts folder, grouping/installing resources, 214-216, 221

Foreground <-> Background command (Color menu), 59

formatting

- removing from text clippings, 220
- text, 101-102

FREF (File Reference) resource, 193-197, 252

G

games, Secret About Box, 228-229, 232-233

General Controls control panel, pattern lists, 78-80

Get info for [file name] command (File menu), 35-36

Get Resource Info command (Resource menu), 28, 40-41

Get Resource Info window, 28-29

grayed-out commands, 278

grayscale, 278

Grid Settings command (DITL menu), 165

grouping resources into System, 214-216, 221

H

hdlg (Dialog Help) resource, 252

heap (System), 283

help, Balloon Help, 273

hex editor, 262-264

hexadecimal numbering system, 278

hfdi (Finder Help) resource, 252

hierarchical menus, 150, 278

hierarchies, 278

hmnv (Menu Help) resource, 252

HogSound 'snd' resource, creating sound files from, 33-36

hot spot tool, 85-86

hot spots (cursors), 85-87, 94, 278

hrct (Rectangle Help) resource, 252

Human Interface Design Guidelines, 278-279

hwin (Window Help) resource, 252

I-J

I-beam, 279

icl4 (4-Bit Color Finder Icon) resource, 252

icl8 (8-Bit Color Finder Icon) resource, 252

ICN# (Black-and-White Finder Icon) resource, 252

ICON (Black-and-White Dialog Icon) resource, 48-60, 253

icon families, 279

- ICON icon, 18
- icon resources, 67
- Icon Size command (icn menu), 58
- icons, 17-18, 48, 279
 - adding to menus, 154
 - applications
 - displaying, 193-197
 - dragging items onto, 193-197
 - editing, 198
 - custom, 64-65, 68
 - dialog
 - color, 53-56
 - editing, 48-52
 - masks, 56-60, 67
 - dialog boxes, 160, 172
 - Finder, 60-65
 - ICON, 18
 - on CD-ROM, 267
 - personal, adding, 244-245, 247
 - small, 65-66
 - keyboard layouts, 137-138
 - 'snd ', 20
- ics# (Black-and-White Small Icon) resource, 253
- ics4 (4-Bit Small Icon) resource, 253
- ics8 (8-Bit Small Icon) resource, 253
- ID numbers, 279
 - fonts, 107-110
 - resources, 41-44
- INIT resource, 253
- Insert New Color command (Resource menu), 180
- Insert New Field(s) command (Resource menu), 91
- Insert New Pattern command (Resource menu), 79
- insertion point, 279
- Inside Macintosh*, 279
- installing
 - editors, 236-238, 246
 - ResEdit, 2-3
 - resources
 - into System, 214-216, 221
 - over existing resources, 4-5
- interfaces, 280
- international codes, 280
- international systems (keyboards), 136-137
- itlb resource, 136
- itlc resource, 137

K

- K (kilobyte), 280
- Kanji, 280
- KBDN (Keyboard Names) resource, 253
- KCHR (Keyboard Characters) resource, 126-140, 253
- KCHR menu commands
 - Convert to Dead Key, 136
 - Edit Dead Key, 135
 - Remove Dead Key, 136
 - Remove Unused Tables, 132-133
 - Uncouple Modifier Keys, 133
 - View As, 133-134
- kcs# (Keyboard Icon, Black and White) resource, 137-138, 253

Zen and the Art of Resource Editing

- kcs4 (Keyboard Icon, 4-Bit)
 - resource, 137-138, 254
- kcs8 (Keyboard Icon, 8-Bit)
 - resource, 254
- kern pairs, 112
- kerning tables, 108
- key codes, 127
- Keyboard control panel, 126
- keyboard layouts, 280
 - customizing, 129-132
 - Dvorak, 276
 - in Finder, 138
 - international systems, 136-137
 - on CD-ROM, 268
 - small icons, 137-138
- keyboard shortcuts
 - Acute accent (Option-E), 134
 - Circumflex (Option-I), 134
 - Command-key equivalents, 148-149, 274
 - editing, 219
 - Copy (Command-C), 32
 - Create New Resource (Command-K), 178
 - Dieresis (Option-U), 134
 - Duplicate (Command-D), 43
 - Eject floppy (Command-Shift-1), 40
 - Get Resource Info (Command-I), 44
 - Grave (Option-`), 134
 - Insert New Color (Command-K), 180
 - New (Command-N), 44
 - Paste (Command-V), 32
 - Save (Command-S), 36
 - Screen capture (Command-Shift-3), 40
 - Select All (Command-A), 54
 - Switch keyboard layouts
 - within script (Command Option Spacebar), 137
 - Switch script (Command-Spacebar), 136
 - tilde (Option-N), 134
- keys
 - dead, 134-136, 140, 275
 - finding on character grid, 140
 - remapping
 - modifier keys, 131-132, 140
 - ordinary keys, 129-130, 140
- keystrokes, Macintosh response
 - process, 127-128
- kilobytes (K), 280
- KMAP resource, 127

L

- lasso tool, 50
- leading, 111
- line tool, 51
- listening to 'snd' resources, 22
- Load Colors command (clut menu), 184
- localization, 280

M

- MACS (Mac System) resource, 254
- maps (files without originating applications), 204-207

- remapping keys
 - modifier keys, 131-132, 140
 - ordinary keys, 129-130, 140
- to editors, changing, 240-242
- masks, 280
 - cursors, 87-88, 94
 - dialog icons, 56-60, 67
- MBAR (Menu Bar) resource, 254
- mcky (Mickey [mouse])
 - resource, 226-228, 254
- mctb (Menu Color Table)
 - resource, 147, 254
- memory
 - adding to Finder, 202-203, 219
 - allocation, 280
 - RAM (Random Access Memory), 282
 - ROM (Read Only Memory), 282
- Menu Bar Patterns utility, 269
- MENU menu commands, 151-152
- MENU resources, 145-154
- menus, 144-154
 - commands, 280-281
 - Finder, editing, 207-209
 - hierarchical, 278
 - separator lines, 282
- message boxes
 - alert boxes, 272
 - dialog boxes, *see* dialog boxes
- metrics (fonts), 111, 278
- modifier keys, remapping, 131-132, 140
- mouse buttons
 - clicking, 274
 - dragging, 276
- Mouse control panel, 226-228
- mouse speed, increasing, 232
- moving resources between files, 31-32, 44

N

- names
 - FOND resource, renaming, 118-119
 - resources, renaming, 43-44
 - sound files, 34-35
 - trash can, renaming, 102
- New command (File menu), 34
- NENT (New FoNT or New Font Numbering Table) resource, 109-110, 254
- numbering dialog-box
 - components, 169-170, 172
- numbers, ID, 279

O

- offline, 281
- opening
 - files without originating applications, 204-207
 - picker windows, 22
 - resources in editor, 22
 - windows, disabling zoom rectangles, 218, 221
- outline fonts, 110-111

P

- paint bucket tool, 50
- Paste command (Edit menu), 32
- 'PAT' (Pattern) resource, 72-73, 254
- PAT menu, Try Pattern command, 73
- PAT# (Pattern List) resource, 78-79, 254
- pattern palette, 51
 - editing, 79
- Pattern Size command (ppat menu), 75
- patterns
 - desktop, 275
 - black-and-white, 72-73, 78-79
 - color, 74-79
 - editing, 79-81
 - on CD-ROM, 267
 - scroll bars, 72-73, 78-79
- pencil tool, 50
- picker windows, 281
 - cicn, 54
 - FKEY, 40-41
 - fmap, 204
 - fmnu, 208
 - MENU, 145
 - opening, 22
 - 'PAT', 72-73
 - PICT, 37-38
 - 'snd', 27
 - vers, 191
- PICT (Picture) resource, 160, 213-214, 254, 281
 - editing, 37-39

- PICT files, 33
 - without originating applications, opening, 204-207
- pictures
 - clippings, 212
 - dialog boxes, 160, 172
- pixels, 281
- playing pig sounds, 6
- pltt (Palette) resource, 185, 187, 255
- pointers, 281
- POST resource, 120-121
- PostScript fonts, 119-121, 281
- PostScript printing language, 281
- ppat (Pixel Pattern) resource, 74-78, 255
- ppat menu commands
 - Pattern Size, 75
 - Try Pattern, 75
- ppt# (Pixel Pattern List) resource, 78-79, 255
- programs, *see* applications
- progress bar (Find File),
 - changing, 213-214, 220
- pushbuttons, 159, 281

Q

- question mark (?), flashing, 8-9
- QuickDraw, 281
 - 32-bit, 272
- quitting Finder, 209-211, 220

R

- radio buttons, 159-160, 281
- RAM (Random Access Memory), 282
- rebooting, 282
- rebuilding desktop, 64, 67-68, 199
- Recent Colors command (Color menu), 59
- Recolor Using Palette command (Color menu), 59-60
- remapping keys
 - modifier keys, 131-132, 140
 - ordinary keys, 129-130, 140
- Remove Colors command (MENU menu), 151
- Remove Dead Key command (KCHR menu), 136
- Remove Icon command (MENU menu), 151
- Remove Unused Tables
 - command (KCHR menu), 132-133
- renaming
 - FOND resource, 118-119
 - resources, 43-44
 - trash can, 102
- Renumber Items command (DITL menu), 164
- renumbering FOND resource, 118-119
- replacing
 - resource contents, 44-45
 - System file, 29-30, 43
- ResEdit
 - components, 15-21
 - installing, 2-3
 - on CD-ROM, 266-268
- resizing patterns, 76-78, 80
- resource files, creating, 44
- resource forks, 282
- Resource menu commands
 - Create New Resource, 54, 178
 - Get Resource Info, 28, 40-41
 - Insert New Color, 180
 - Insert New Field(s), 91
 - Insert New Pattern, 79
- resource trees, 278, 282
- resources, 12-14
 - actb (Alert Color Table), 250
 - acur (Animated Cursor), 89-91, 250
 - ALRT (Alert Info), 167-171, 250
 - BNDL (Bundle Resource), 193-198, 250
 - cicn (Color Icon), 53-56, 250
 - clut (Color Look-Up Table), 178-187, 250
 - CODE, 250
 - creating, 67
 - crsr (Color Cursors), 88-89, 250
 - CURS (Black-and-White Cursors), 84-88, 250
 - dctb (Dialog Color Table), 251
 - DITL (Dialog Item List), 158-167, 251
 - DLOG (Dialog), 167-171, 251
 - drag, 212
 - DSAT (Dialog Satellite), 231, 251
 - duplicating, 43
 - FKEY (Function Key), 39-41, 229-230, 251, 277

- fmap (Finder Mapping),
204-207, 219, 251
- fmnu (Finder Menu),
207-211, 219, 251
- FOND (Font Definition),
107-108, 251
 - editing, 114-117
 - renaming/renumbering,
118-119
- FONT, 107, 251
 - adding to FOND re-
sources, 122-123
 - editing, 112-114
- FREF (File Reference),
193-197, 252
- grouping/installing into
System, 214-216, 221
- hdlg (Dialog Help), 252
- hfdi (Finder Help), 252
- hmnu (Menu Help), 252
- hrct (Rectangle Help), 252
- hwin (Window Help), 252
- icl4 (4-Bit Color Finder Icon),
252
- icl8 (8-Bit Color Finder Icon),
252
- ICN# (Black-and-White
Finder Icon), 252
- ICON (Black-and-White
Dialog Icon), 48-60, 253
- icon, 67
- ics# (Black-and-White Small
Icon), 253
- ics4 (4-Bit Small Icon), 253
- ics8 (8-Bit Small Icon), 253
- IDs, 41-44
- in files, 22, 30-31
- INIT, 253
- installing over existing
resources, 4-5
- itlb, 136
- itlc, 137
- KBDN (Keyboard Names),
253
- KCHR (Keyboard
Characters), 126-140, 253
- kcs# (Keyboard Icon,
Black-and-White), 137-138,
253
- kcs4 (Keyboard Icon, 4-Bit),
137-138, 254
- kcs8 (Keyboard Icon, 8-Bit),
254
- KMAP, 127
- MACS (Mac System), 254
- mapping to other editors,
240-242
- MBAR (Menu Bar), 254
- mcky (Mickey [mouse]),
226-228, 254
- mctb (Menu Color Table),
147, 254
- MENU, 145-154
- moving between files,
31-32, 44
- NFNT (New FoNT or New
Font Numbering Table),
109-110, 254
- on CD-ROM, 266
- opening in editor, 22
- 'PAT ' (Pattern), 72-73, 254
- PAT# (Pattern List),
78-79, 254
- PICT (Picture),
160, 213-214, 254, 281
 - editing, 37-39

pltt (Palette), 185, 187, 255
 POST, 120-121
 ppat (Pixel Pattern), 74-78, 255
 ppt# (Pixel Pattern List), 78-79, 255
 renaming, 43-44
 replacing contents, 44-45
 RMAP (Resource Mapping), 240-242, 246-247, 255
 ROv# (ROM Override), 255
 sfnt (Spline Font), 110-111, 255
 editing, 117-118
 SICN (Small Icon), 66, 255
 SIZE, 203, 255
 'snd ' (Sound), 20-21, 255
 "camera click" sound, adding to Sound control panel, 26-30
 creating sound files from, 33-36
 sound, 20-21
 'STR ' (String), 98-99, 256
 STR# (String list), 99-100, 256
 styl (Style), 101-102, 256
 TEXT, 101-102, 256
 TMPL (Template), 238-240, 246, 256
 vers (Version), 190-193, 197-198, 256
 working process, 14-15
 restoring System file, 8-9
 RMAP (Resource Mapping) resource, 240-242, 246-247, 255

ROM (Read Only Memory), 282
 ROv# (ROM Override) resource, 255
 RSED creator type, 35
 rsrc file type, 35

S

safety guidelines
 color palettes, 187
 dialog boxes, 173
 Finder, 221
 general work guidelines, 23
 keyboards, 140
 menus, 154
 modified software, 199
 System file, 45
 text, 102
 Save command (File menu), 36
 screen-shot creators, changing default, 229-230, 233
 screens
 splash, 16
 see also windows
 Script Manager, 136-137, 282
 scroll bar patterns, 72-73, 78-80
 Secret About Box, 228-229, 232-233
 Select All command (Edit menu), 54
 Select Item command (DITL menu), 165
 selection rectangle tool, 50
 separator lines, 282
 sfnt (Spline Font) resource, 110-111, 255
 editing, 117-118

Zen and the Art of Resource Editing

- shibboleth, 282
- Show All Items command
 - (DITL menu), 165
- Show Item Numbers command
 - (DITL menu), 165
- SICN (Small Icon) resource,
 - 66, 255
- SIZE resource, 203, 255
- small icons, 65-66
 - keyboard layouts, 137-138
- 'snd ' (Sound) resource, 20-21, 255
 - "camera click" sound, adding
 - to Sound control panel, 26-30
 - creating sound files from, 33-36
 - listening to, 22
- 'snd ' icon, 20
- snd menu, Try Sound command, 27
- software, *see* applications
- Sound control panel, adding
 - "camera click" sound to, 26-30
- sound files, creating from 'snd ' resources, 33-36
- sound resources, 20-21
- sounds
 - clippings, 212
 - on CD-ROM, 268
 - pigs, playing, 6
- splash screen, 16
- Standard 16 Colors command
 - (Color menu), 59
- Standard 16 Grays command
 - (Color menu), 59
- Standard 256 Colors command
 - (Color menu), 59
- Standard 4 Grays command
 - (Color menu), 59
- standard file dialog boxes, 282
- startup dialog box, changing
 - Welcome to Macintosh text, 230-231, 233
- static text (dialog boxes), 160
- stationery files, giving
 - applications icons for, 196-198
- stop sign icon
 - color, 54-56
 - editing, 48-52
- 'STR ' (String) resource, 98-99, 256
- STR# (String list) resource,
 - 99-100, 256
- strings, *see* text
- styl (Style) resource, 101-102, 256
- style tables, 108
- suitcases, 282
- System, grouping/installing
 - resources, 214-216
- System file
 - backing up, 3
 - copying for editing, 26-27, 42-43
 - replacing, 29-30, 43
 - restoring, 8-9
- System Folder, 283
- System heap, 283
- System palette, 177

T

tables

- actb (Alert Color TaBle), 171
- clut (color look-up table), 176-177
 - loading from System or ROMs, 187
- dctb (Dialog Color TaBle), 171
- FATs (Font Association Tables), 108, 115-117
- kerning, 108
- style, 108

templates, 238-240

- on CD-ROM, 269-270

text, 98-102

- clippings, 212, 220
- editable (dialog boxes), 160, 172
- Finder, 216-217
- static (dialog boxes), 160

TEXT files, 33

- without originating applications, opening, 204-207

TEXT resource, 101-102, 256

text view, 283

TMPL (Template) resource, 238-240, 246, 256

Toolbox, 283

tools area of editors, 50-51

- eyedropper tool, 55-56
- hot spot tool, 85-86

trash can icon, 61-62

- renaming, 102

TrueType fonts, 283

Try Pattern command

- PAT menu, 73
- ppat menu, 75

Try Pointer command (CURS menu), 88

Try Sound command (snd menu), 27

ttro file type, 33

txtxt creator type, 32

typefaces, 283

- see also* fonts

U

Uncouple Modifier Keys

- command (KCHR menu), 133

Use Color Picker (MENU menu), 152

user items (dialog boxes), 160

utilities

- ClickChange (Dubl-Click Software), 92
- Color Cursor, 88-89
- CursorAnimator, 92, 268-269, 275
- on CD-ROM, 268-269

V

verifying files, 243, 247-248

vers (Version) resource, 190-193, 197-198, 256

version numbering schemes, 283

Zen and the Art of Resource Editing

View As command
DITL menu, 165-166
KCHR menu, 133-134

W-Z

watch cursor, 283
 editing, 93-94
Welcome to Macintosh dialog
 box, changing text, 230-231,
 233
windows
 close box, 274
 Get Resource Info, 28-29
 picker, 281
 cicn, 54
 fmap, 204
 fmnu, 208
 MENU, 145
 opening, 22
 'PAT ', 72-73
 'snd ', 27
 vers, 191
Zipple extension, 269
zoom rectangles, disabling, 218,
 221

You've mastered ResEdit. Now it's time for AppleScript.

**Let BMUG help you speed up,
customize, and automate your Mac!**

The Tao of AppleScript is the premier guide to Macintosh scripting, and it comes to you from the world's largest Macintosh user group. Journey into the ever-expanding world of customizing applications and automating tedious tasks. This friendly book features expanded coverage of AppleScript basics, new information on the Scriptable Finder, and more hands-on examples. With *The Tao of AppleScript* and the enclosed software, you can control your Mac like never before!



Everything you need to begin
AppleScripting



Clear, step-by-step instructions for
creating effective scripts



Example scripts included enable you to
get started fast

The Tao of AppleScript, Second Edition

by Derrick Schneider with Hans Hansen & Tim Holmes

\$29.95 US

ISBN: 1-56830-115-4



Look for *The Tao of AppleScript*
at your favorite book retailer!



"*The Tao of AppleScript* is really first rate.
If you want to learn AppleScript, this is
where to start. End of story!"

Don Crabb, columnist, *MacWEEK*

"The best way I know of to get
started...."

David Ramsey, *MacWEEK*

"An excellent introduction to the
language....The style is comforting and
aimed at every Mac user out there,
from novice to expert."

Chris Oakes, *Computer Currents*

PLUG YOURSELF INTO...



THE MACMILLAN INFORMATION SUPERLIBRARY™

**Free information and vast computer resources from
the world's leading computer book publisher—online!**

FIND THE BOOKS THAT ARE RIGHT FOR YOU!

A complete online catalog, plus sample chapters and tables of contents give you an in-depth look at *all* of our books, including hard-to-find titles. It's the best way to find the books you need!

- **STAY INFORMED** with the latest computer industry news through our online newsletter, press releases, and customized Information SuperLibrary Reports.
- **GET FAST ANSWERS** to your questions about MCP books and software.
- **VISIT** our online bookstore for the latest information and editions!
- **COMMUNICATE** with our expert authors through e-mail and conferences.
- **DOWNLOAD SOFTWARE** from the immense MCP library:
 - Source code and files from MCP books
 - The best shareware, freeware, and demos
- **DISCOVER HOT SPOTS** on other parts of the Internet.
- **WIN BOOKS** in ongoing contests and giveaways!

TO PLUG INTO MCP:

GOPHER: gopher.mcp.com

FTP: ftp.mcp.com

WORLD WIDE WEB: <http://www.mcp.com>



GET CONNECTED
to the ultimate source
of computer information!

The MCP Forum on CompuServe

Go online with the world's leading computer book publisher!
Macmillan Computer Publishing offers everything
you need for computer success!

Find the books that are right for you!

A complete online catalog, plus sample
chapters and tables of contents give
you an in-depth look at all our books.
The best way to shop or browse!

- Get fast answers and technical support for
MCP books and software
- Join discussion groups on major computer
subjects
- Interact with our expert authors via e-mail
and conferences
- Download software from our immense
library:
 - Source code from books
 - Demos of hot software
 - The best shareware and freeware
 - Graphics files



Join now and get a free CompuServe Starter Kit!

To receive your free CompuServe Introductory Membership, call 1-800-848-8199 and ask for representative #597.

The Starter Kit includes:

- Personal ID number and password
- \$15 credit on the system
- Subscription to *CompuServe Magazine*



Once on the CompuServe System, type:

GO MACMILLAN

for the most computer information anywhere!

 **CompuServe**

CD-ROM at a Glance

The CD-ROM that comes with this book contains everything you need to get into the heart of your Mac and redesign it to fit your personality.

- **ResEdit 2.1.3.** This piece of software from Apple Computer is the basis of this entire book—so this is a good place to start on your road to customizing your Mac. This version is Power Mac-compatible and updated to work with System 7.5.
- **Zen ResEdit Apple Guide.** If you're using System 7.5 on your Macintosh, you can take advantage of this online guide to using ResEdit. With the file on this CD-ROM installed on your hard drive, you have access to all the step-by-step instructions in this book. So, if you're using ResEdit on a PowerBook on the road or just want to quickly remember how to edit a keyboard map, you won't need to go any further than your Help menu.
- **Thousands of resources.** A good beginning for realizing your dreams of a completely customized Macintosh. You can use the thousands of icons, patterns, cursors, and other resources in this collection to supplement your own artistic skills and creative desires. All these resources are organized by their type and author so you can quickly find your way to what you need.
- **Utilities.** In addition to ResEdit, there are hundreds of great shareware and freeware utilities on the CD-ROM for customizing your Mac. These utilities enhance what you can do with resources and let you explore whole new areas of your Macintosh. They are all organized by subject and have documentation.
- **ResEdit extensions.** Dozens of "plug-in" editors and templates expand what you can do with ResEdit by letting you access even more resource types.
- **CD Highlights.** Feeling overwhelmed by all the stuff on the CD-ROM? Let us guide you to our favorite resources and utilities with a folder of aliases that take you right to them.
- **Book examples.** Follow along with the text of the book, exploring the same resources used during the creation of the book.

Optional Floppy Disk

If you don't have access to a CD-ROM drive, you can call BMUG for a copy of ResEdit on a floppy disk. This 800K floppy contains ResEdit 2.1.3, the examples used in the book, the Zen ResEdit Apple Guide, and a few cool resources to play with.

You can contact BMUG at:

Mail: BMUG, 1442A Walnut Street, #62, Berkeley, CA 94709-1496

Fax: (510) 849-9026

Voice: (510) 549-2684

(USA Only): (800) 776-2684



EXPANDED FOURTH EDITION

"No true raging, butt-kicking, inexorable Macintosh evangelist should be without this book."

— Guy Kawasaki, Apple Fellow,
author of *How to Drive Your
Competition Crazy*

People don't all look or work the same, so why should their Macs? Let your individuality shine through! Show it off with ResEdit—the fun way to customize the look and feel of your Mac. But it's not just fun and games—you can even use ResEdit to tailor your computer to work more efficiently.

Developed by Apple, ResEdit provides point-and-click access to the heart of the Macintosh—its resources. From icons to menus, resources are the components inside everything on your Mac.

LEARN TO CREATE AND CUSTOMIZE:

- Icons
- Patterns
- Cursors
- Dialog boxes
- Menus
- Fonts
- Color palettes
- Keyboard layout
- Application resources
- The Finder and System
- And more!

THE ZEN CD-ROM INCLUDES:



- A FULL VERSION OF
ResEdit 2.1.3
- SYSTEM 7.5 APPLE GUIDE
FOR ONLINE ResEdit HELP
- ZILLIONS OF RESOURCE FILES
- USEFUL ResEdit
EXTENSIONS
- TONS OF GREAT UTILITIES
TO ENHANCE YOUR MAC

"Everything you need to
know about warming
your soul through Macintosh."

— Cliff Stoll, author of
The Cuckoo's Egg and *Silicon
Snake Oil*

Derrick Schneider and **Hans Hansen** have a long association with BMUG and are also the authors of *The Tao of AppleScript*.

\$30.00 USA / \$40.95 CAN /
£27.95 Net UK (inc of VAT)



Category: General Macintosh
User Level: Beginner—Intermediate

REQUIREMENTS:
Mac System 7 or higher
4 MB of RAM recommended
CD-ROM drive

