

YOUR NO SWEAT, STEP-BY-STEP, TICKET TO THE INTERNET!

Adam C. Engst

Includes an electronic version of the bestselling, #1 book on the Internet, *Internet Starter Kit for Macintosh, 3rd Edition* by Adam C. Engst



Features the exciting multimedia CD-ROM "Internet Interactive"

Simply Amazing Internet

for Macintosh®

Simply Amazing Internet

for
Macintosh

Simply Amazing Internet for Macintosh

Adam C. Engst



Hayden
Books

Simply Amazing Internet for Macintosh

©1995 Hayden Books, a division of Macmillan Computer Publishing

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: 95-777-37

ISBN: 1-56830-230-4

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

The Hayden Books Team

Publisher

David Rogelberg

Acquisitions Editor

Karen Whitehouse

**Manager of Product
Development**

Pat Gibbons

Development Editor

Brian Gill

Marketing Manager

Andy Roth

Marketing Coordinator

Meshell Dinn

Publishing Coordinator

Rosemary Lewis

Interior Designers

Fred Bower, Barbara Kordesh,
Kevin Spear

Illustrator

Kevin Spear

Cover Designer

Jay Corpus

Manufacturing Coordinator

Paul Gilchrist

Production Manager

Kelly D. Dobbs

Production Team Supervisor

Laurie Casey

Graphics Coordinator

Dennis Clay Hager

Graphic Image Specialists

Clint Lahnen, Dennis Sheehan

Production Analysts

Angela Bannan, Bobbi
Satterfield, Mary Beth Wakefield

Production

Dan Caparo, Angela Calvert,
Aleata Howard, Erika Millen,
Gina Rexrode, Karen Walsh

Indexer

Brad Herriman

Composed in Palatino and Futura

License Information

HAYDEN'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. HAYDEN'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 TO 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 HAYDEN'S LICENSOR(S), AND THEIR DIRECTORS, OFFICERS, EMPLOYEES OR AGENTS (COLLECTIVELY HAYDEN'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 HAYDEN'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. Hayden'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.

To Our Readers

Dear Friend,

Thank you on behalf of everyone at Hayden Books for choosing *Simply Amazing Internet for Macintosh* to enable you to explore the Internet. The Internet is the fastest growing, and most exciting, aspect of our modern society. Without the right guide, participating on the Internet can be difficult. We have carefully crafted this book and CD to enable you to both learn about the Internet and actually get on it. We wish you the best in your explorations and hopefully, we'll meet you there soon!

What you think of this book is important to our ability to better serve you in the future. If you have any comments, no matter how great or small, we'd appreciate you taking the time to send us email or a note by snail mail. Of course, we'd love to hear your book ideas.

Sincerely yours,



David Rogelberg
Publisher, Hayden Books and Adobe Press

You can reach Hayden Books at the following:

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

Email addresses:

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

Dedication

To my lovely and talented wife, Tonya, without whose patience and writing skills I would certainly have gone stark raving mad long ago.

About the Author

Adam C. Engst

Adam C. Engst is the editor and publisher of *TidBITS*, a free electronic newsletter distributed weekly on the worldwide computer networks to over 150,000 readers. After graduating from Cornell University with a double-major in Hypertextual Fiction and Classics, he worked as an independent consultant in Ithaca, New York, where he started *TidBITS* in April of 1990. He now lives in Western Washington with his wife, Tonya, and cats, Tasha and Cubbins, but seems to spend most of his time corresponding via email with friends and associates around the globe.

After writing the first edition of *Internet Starter Kit for Macintosh* (released in September of 1993), he was somehow rooked into coauthoring *Internet Explorer Kit for Macintosh* with Bill Dickson, and the first edition of *Internet Starter Kit for Windows* with Cory Low and Mike Simon. These titles were published in April of 1994, and the second edition of *Internet Starter Kit for Macintosh* came out in August of 1994. In March of 1995, the second edition of *Internet Starter Kit for Windows* was published, again in collaboration with Cory and Mike. The third edition of *Internet Starter Kit for Macintosh* came out in June of 1995.

Like anyone who attempts to condense the immensity of the Internet into a single book, he is certifiably crazy. His favorite quote (which reportedly comes from Alan Kay, an Apple Fellow and Xerox PARC alumnus) is: "The best way to predict the future is to invent it."

Acknowledgments

Thanks

Many people have helped me throughout the writing of the various editions of *Internet Starter Kit for Macintosh*, on which this text is based. I especially want to thank my wife, Tonya, who continues to help more and more.

I'd also like to thank all the readers of *TidBITS* because they have kept me going these last five years. Finally, thanks to the many readers and supporters of the previous editions of *Internet Starter Kit for Macintosh*. Word of mouth goes a long way, and the book wouldn't have lasted nearly this long without such support from so many individuals in the Internet community.

If I've forgotten anyone, my sincere apologies. Please accept my absent-minded thanks.

Surrogate Editors

Thanks are due to the many people who helped edit portions of *Internet Starter Kit for Macintosh* and to those who either provided me with information or helped test the ISKM Installer. I wish only that I had room to give everyone the full credit they deserve.

Alan Oppenheimer, Alwin Li, Amanda Walker, Amir Rosenblatt, Andrew Green, Barbara Maxwell, Bill Cockayne, Bill Dickson, Bill Gram-Reefer, Bill Lipa, Brian Hall, Brian Lichorowic, Cary Lu, Chad Magendanz, Chuck Shotton, Chuq Von Rospach, Cory Low, Dave Winer, David Thompson, Draper Kauffman, Drummond Reed, Erik Speckman, Franklin Davis, Geoff Duncan, Glenn Fleishman, Gordon Watts, Guy Kawasaki, Guy Stevens, Harry Morris, Herb Effron, Ian Brown, Igor Livshits, James Skee, Jeff Needleman, Jeff Steele, Jim Matthews, Jim O'Dell, John Baxter, John Hardin, John Mah, John Norstad, John Werner, Jonathan Lundell, Jud Spencer, Katie Hudson, Kee Nethery, Ken Landau, Ken Stuart, Larry Blunk, Laurent Humbert, Len MacDonell, Leonard Rosenthol, Mark H. Anbinder, Mark McCahill and the Gopher Team, Mark Verber, Mark Williamson, Meshell Dinn, Michael Tardiff, Michael Wellman, Mikael Hansen, Mike Goode, Mike Simon, Paul Celestin, Pete Resnick, Peter Broadribb, Peter Fraterdeus, Peter Lewis, Quinn, Richard Ford, Rick Watson,

Roger Brown, Roger Jacobs, Ross Scott Rubin, Roy Wood, Roz Ault, Rusty Tucker, Sandro Menzel, Scott Gruby, Sky Dayton, Steve Dagley, Steve Dorner, Steve Michel, Steve Mulder, Steve Worona, Werner Uhrig, and Will Mayall.

Software Thanks To...

Although I don't have room to individually acknowledge the talented programmers of all the software included on the CD, I do want to thank those who graciously allowed me to include their programs in the ISKM Installer, including the following:

Leonard Rosenthol, and all the folks at Aladdin Systems, for StuffIt InstallerMaker and StuffIt Expander

Garry Hornbuckle and the MacTCP team at Apple Computer, for MacTCP

Larry Blunk and Merit Network, for MacPPP

Amanda Walker and InterCon Systems, for InterSLIP

Peter Lewis, for Anarchie

Peter Lewis, again, and Quinn "The Eskimo!", for Internet Config

Steve Dorner and Qualcomm, for Eudora

John Hardin of EINet, for MacWeb

Other Invaluable Aid Provided By...

Liam Breck, for helping set up the Info-Mac Archive mirror software at **ftp.tidbits.com**

Ed Morin, Ralph Sims, Michael Tardiff, and Scott Anderson of Northwest Nexus for technical assistance above and beyond the call of duty and for providing my Internet connections

Bill Dickson and Rob Furr, for helping to translate *Internet Starter Kit for Macintosh* into HTML

Jim Pickrell and Harry Wiland, for persevering with the CD

And finally, Geoff Duncan, once again, for creating the *Internet Starter Kit for Macintosh* Web page and for helping to test the CD

Contents at a Glance

Part I: Introduction and History	1
1 Internet Interactive Installation	3
2 Welcome	19
3 Why Is the Internet Neat?	23
4 What Is the Internet?	29
5 The Internet Beanstalk	51
Part II: Internet Foundations	65
6 Addressing & URLs	67
7 Email Basics	81
8 Usenet News Basics	109
9 TCP/IP Internet Services	125
10 File Formats	145
Part III: Full Internet Access	153
11 Making Connections	155
12 MacTCP-based Programs	181
Glossary	223
Index	243

Table of Contents

Part I Introduction and History	1
1 Internet Interactive Installation	3
Quick Start	3
ISKM Installer	4
How to Install	5
Installation Option Details	10
Internet Interactive CD?	12
What's on the CD?	12
Talk to Adam	13
Step-by-Step	13
Read the Book	13
Best of Internet	13
CD Art Tour	14
About Leonardo Multimedia	14
Connecting with EarthLink Network	14
EarthLink Network TotalAccess USA 800	14
EarthLink Network TotalAccess Southern California	14
EarthLink Network TotalAccess USA	14
Other Internet Access Providers	15
Setting up an Account	15
There's More...	15
ftp.tidbits.com	16
ISKM Home Page	16
On to the Small Talk	17
2 Welcome	19
Who Are You?	19
What Do You Need?	20
Who Am I?	21
Changes	22
3 Why Is the Internet Neat?	23
Electronic Mail	24
Discussion Groups	24
Software	25
Information at Your Electronic Fingertips	25

The Lemming Factor	26
The Internet Is What You Make of It	27
Champing at the Bit	28
4 What Is the Internet?	29
Size.....	29
People.....	30
Machines.....	30
Information	31
Geographic Size	31
People—Doing What They Do Best.....	32
Hardware.....	34
Computers.....	34
Networks	35
Modems and Phone Lines	36
Software	39
Information	41
The Internet Post Office.....	42
Electronic Mail	44
Usenet News	45
Telnet	46
File Transfer Protocol (FTP).....	47
WAIS	48
Gopher	48
World Wide Web	49
Summing Up	50
5 The Internet Beanstalk.....	51
Cold War Network.....	51
Gateways	53
Usenet	54
BITNET	55
NSFNET	55
NREN	56
Who Pays?	57
Politics	59
The Future	60
Past and Future—Next, the Present	63

Part II Internet Foundations	65
6 Addressing & URLs	67
Addressing	67
Top-level Domains	68
Mid-level Domains	70
Machine Name	71
Userid	72
Punctuation	73
Alternative Addresses	74
URLs	75
What are URLs?	75
URL Construction	76
Using URLs	78
Weird Characters	79
A Self-Addressed Internet... ..	80
7 Email Basics	81
Email Construction	81
Signature	81
Body	82
Header	87
Bounces	90
Using Email Programs	91
Reading Email	91
Navigating and Managing Email	92
Replying to Email	93
New Email	94
Finding People	95
Friends	96
Acquaintances	97
Mailing Lists	97
Moderated vs. Unmoderated	99
Individual Messages vs. Digests	100
Mailing List Managers	101
Neither Rain, Nor Snow... ..	107
8 Usenet News Basics	109
Usenet Plumbing	110
Newsgroup Innards	111
Message Construction	113
The Newsgroup Stork	115

Using Usenet	117
Subscribing to Groups	117
Reading Articles	117
Navigation	118
rot13	120
Extracting Binary Files	120
Replying to Articles	121
Follow-ups	121
Posting an Article	122
ClariNet	123
Enough Usenet	124
9 TCP/IP Internet Services	125
FTP	125
FTP Manners	126
FTP Clients	127
Archie	128
Telnet Usage	130
IRC	131
MUDs	132
WAIS	133
Gopher	137
Veronica	139
Jughead	140
World Wide Web	140
Wrapping Up	143
10 File Formats	145
ASCII Encoding	146
BinHex	147
Compression Formats	148
Stuffit 3.0	149
Compact Pro	150
Self-Extracting Archives	150
Format's Last Theorem	151
Part III Full Internet Access	153
11 Making Connections	155
MacTCP	155
MacPPP	157

Installation and Setup	158
Basic Usage	165
MacPPP Details	166
But What If... ..	166
Troubleshooting Your Connection	166
Reporting Problems	167
MacTCP Q & A	169
Domain Name Server Errors	171
MacPPP Q & A	175
Moving On	179
12 MacTCP-based Programs	181
Internet Config	181
Internet Config: Step-by-Step	183
Internet Config Details	186
Eudora	186
Eudora: Step-by-Step	188
Eudora Details	195
NewsWatcher	195
NewsWatcher: Step-by-Step	196
NewsWatcher Details	202
Anarchie	202
Anarchie: Step-by-Step	202
Anarchie Details	205
MacWeb	206
MacWeb: Step-by-Step	207
MacWeb Details	211
Netscape Navigator	212
Netscape Step-by-Step	212
Netscape Details	217
Helper Applications	218
WWWrapping Up	222
Glossary	223
Index	243

Introduction

Before anything else, please believe me when I say that this book is not a manual to be ignored after you've ripped the CD from the cover and stuffed it in your Macintosh. Using the Internet is not like using a word processor. There are some quirks and confusions that I explain, but they may frustrate you to no end if you don't bother to read the sections about how to use the CD or configure the software for using the Internet.

That said, I want to tell you a little about how I designed *Simply Amazing Internet for Macintosh*, because I think it will help you make the most of the book in the limited time that we all have. As with any broad subject, some parts of the Internet won't interest you in the slightest, so I've tried to keep the contents of this book focused on the background information you need to know and the essential instructions for using the most popular programs. So, let's look quickly at each of the parts that make up this book, and I'll tell you what to expect.

Part I: Introduction and History

First off, since I know you're champing at the bit, the first chapter walks you through installing the CD on your Mac and gives you a brief look at the main features of the CD. After chapter 1, this part stays fairly general, introducing you to the book and to the Internet, and trying to offer a way of looking at the world through Internet-colored glasses. Chapter 2, "Welcome," serves as the introduction. Chapter 3 seeks to answer the question, "Why Is the Internet Neat?" The fourth chapter, "What Is the Internet?," tackles a more difficult issue and attempts to explain just what the Internet is. Next, Chapter 5, "The Internet Beanstalk," traces a line down the history of the Internet, branching off to look briefly at other topics of potential interest.

Part II: Internet Foundations

Part II is required reading for novices because it covers network foundations in terms of basic usage and social customs. It's important because it provides much of the Internet's community knowledge. If you want to know how email is constructed, for example, or when you should avoid using FTP, you should read

this part. Community knowledge can be difficult to pick up without spending a lot of time on the Internet, so these chapters can save you a lot of time and trouble.

Chapter 6, “Addressing & URLs” covers the basics of addressing and focuses tightly on Uniform Resources Locators, or URLs, which are used to provide a coherent method of identifying Internet resources such as Web pages and files available via FTP. Chapter 7, “Email Basics,” looks at email usage and mailing lists, and Chapter 8, “Usenet News Basics,” is devoted exclusively to Usenet news. Chapter 9 describes “TCP/IP Internet Services,” those being services that require a full Internet connection, such as Telnet, FTP, WAIS, Gopher, and the World Wide Web. Finally, Chapter 10, “File Formats,” discusses the main file formats you find on the Internet.

Part III: Full Internet Access

Chapters 11 and 12 of Part III look in depth at the most powerful and flexible method of connection to the Internet—the MacTCP connection. Although the other methods of connection are important and useful, nothing competes with the range and quality of software available for Macs connected to the Internet and running MacTCP. I felt so strongly about helping readers get a full MacTCP connection to the Internet that Hayden Books licensed MacTCP from Apple and included it on the CD. Other programs that you need to immediately start using a MacTCP connection via your modem are also included.

Chapter 11 tells you about MacTCP, the low-level program that makes everything possible. It then introduces MacPPP and walks you through configuring MacPPP for EarthLink Network. As we all know, there’s no way to guarantee instant success in this world, so the “Troubleshooting Your Connection,” section in chapter 11 tips the scales in your favor against the entropy of the universe. At least, that’s how I like to look at it.

Although chapter 11 will get you on the Internet with a MacTCP connection, nothing happens until you run a MacTCP-based program. That’s what I cover in chapter 12, starting with “Email” and “Usenet News,” and continuing on to “FTP” and “World Wide Web.” In the “Email” section, I discuss the free Eudora, the best Internet email program available. The “Usenet News” section concentrates on another excellent program, John Norstad’s free NewsWatcher. For “FTP,” I look at Anarchie, a \$10 shareware program from Peter Lewis, that may be the best FTP program for any type of computer. And finally, in the “World Wide Web” section, I discuss two popular Web browsers, MacWeb and Netscape Navigator, along with the various helper applications that you use for

XXII *Simply Amazing Internet for Macintosh*

types of information like sounds and movies that the Web browsers don't understand.

Welcome to *Simply Amazing Internet for Macintosh*, welcome to the Internet, and please, make yourself at home.

ADAM C. ENGST

August, 1995

Part I



Introduction and History

In this part of *Simply Amazing Internet for Macintosh*, I introduce the CD and myself and give you a look at what the Internet is, why it's so neat, and where it came from. These five chapters convey the proper mindset for thinking about the Internet, a mindset without which you may find the Internet an overwhelming place.

First, in chapter 1, I walk you through installing the CD and give you a brief tour of its main features. Chapter 2, "Welcome," starts out slowly to allow us to introduce ourselves to one another. It also lays out the basic requirements for using the Internet. Chapter 3, "Why Is the Internet Neat?," talks about what makes the Internet a special place, and chapter 4, "What Is the Internet?," supports that by attempting to define the Internet.

Chapter 5, "The Internet Beanstalk," provides a brief look at the past and future of the Internet, although if you don't like reading about history, feel free to skip it. If you find yourself condemned to repeat history at some later time, though, don't blame me, since chapter 5 provides the background that you need to understand why the Internet is the way it is and how it works politically.

Chapter 1



Internet Interactive Installation

The CD that comes with *Simply Amazing Internet for Macintosh* contains all the software you need to access the Internet, along with movies that let you see how the software is configured, an art tour, a Best of the Internet section, the entire text—complete with screenshots—of *Internet Starter Kit for Macintosh, Third Edition*, and more.

Quick Start

If you're the impatient type, you probably want to get started already, having wasted valuable seconds reading the above paragraph. There are three basic steps you must follow to get started:

1. Insert the CD and run the ISKM Installer. Along with installing the software necessary to access the Internet, such as MacTCP and MacPPP, the ISKM Installer's standard installation option installs QuickTime 2.0 for you, and you need QuickTime to view the movies from the CD.
2. Double-click on the icon labeled "Start Here" to launch the Internet Interactive application and start browsing around in it.
3. Call EarthLink Network at 213-644-9500 to set up your Internet account so you can follow links out of the CD to the Internet (and use the Internet on your own).

Note

If you live in Los Angeles (in the 213, 310, or 818 area codes), you may wish to consider, instead of EarthLink Network, another Internet provider called Leonardo Internet. Run by the same folks who created the Internet Interactive CD-ROM, Leonardo Internet offers unlimited Internet access for \$22 per month (paid quarterly). Look in the About Leonardo folder on the CD for a Leonardo ISKM Installer that's customized for Leonardo (it installs exactly the same stuff other than customized MacTCP Prep and PPP Preferences files). Using the Leonardo ISKM Installer is exactly the same as the standard ISKM Installer; all that's different in the installer are the names of the options of some of the descriptions. Once you've installed the software, call Leonardo Internet at 310-395-5500 for details and to sign up.

Now, for those of you with a bit more patience, read on for the details of the installation process.

ISKM Installer

I designed the ISKM Installer to be as easy as possible to use and tried to think of the different things you might want to do, which is why I created several different installer options. To install any or all of the main Internet programs included in the installer (as well as QuickTime, which is necessary to play the movies on the CD), double-click the ISKM Installer icon and follow the instructions from that point. See the following text for a detailed list of what's installed by each option in the ISKM Installer.

The ISKM Installer includes:

- MacTCP 2.0.6, which enables your Macintosh to run MacTCP-based Internet programs.
- MacPPP 2.0.1, which enables you to use a modem to connect to a PPP account at your Internet access provider.
- QuickTime 2.0, which enables you to play the movies on the CD.
- Anarchie 1.5, the best FTP client for any computer.
- Eudora 1.5.2, a powerful program that enables you to send and receive Internet email.
- Internet Config 1.1, which centralizes preferences like your email address for a number of Internet programs.
- MacWeb 1.00A3.2, an excellent browser for the World Wide Web that takes you to the Internet Starter Kit for Macintosh Home Page by default.
- StuffIt Expander 3.5.2, which can debinhex and expand the files you download from the Internet.
- Bookmarks for retrieving the latest versions of all the major Internet programs.

The ISKM Installer will not overwrite existing files other than QuickTime, QuickTime PowerPlug, MacTCP Prep, or PPP Preferences (all of which you must have installed properly for the various instructions to work—if you want to replace a file such as MacTCP or Config PPP, you must first move it out of the System Folder manually. This is imperative if you have installed Apple's System 7.5 (which comes with MacTCP), if you are switching from SLIP to PPP, or if you have already used some of this software previously and wish to take advantage of the included preferences and configurations.

How to Install

As with most installers, using the ISKM Installer is a piece of cake. First, restart your Macintosh while holding down the Shift key. This process disables all of your extensions until the next time you restart. Although not absolutely necessary, it's a good idea to do this to ensure that anti-virus software or other extensions don't interfere with the installation process (which they sometimes do).

Insert the CD in your drive, and double-click on the Internet Interactive icon. When the window appears, double-click on the file called ISKM Installer. You should see the ISKM Installer splash screen. Click on the Continue button. Next, the ISKM Installer presents you with a screen full of text that describes what the various options available in the ISKM Installer do, and offers other useful information. Read this information, and if you want, save or print it by clicking the Save or Print buttons. Then click the Continue button. The ISKM Installer then presents you with the Standard Install dialog box (see figure 1.1).

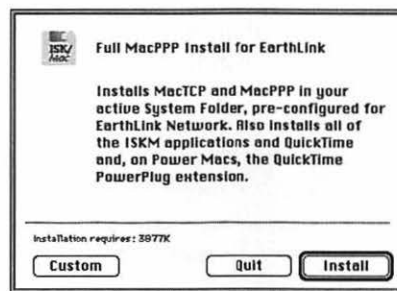


Figure 1.1 ISKM Installer Standard Install dialog.

Here's where you must make your first decision.

- If you want to install everything necessary to use the CD and the Internet, select the standard **Full MacPPP Install for EarthLink**, which installs all the applications and MacTCP and MacPPP, along with QuickTime 2.0 and, on Power Macs, QuickTime

PowerPlug. A partially configured PPP Preferences file containing some modem configurations is also installed for you. Contact EarthLink for the information necessary to finish configuring MacPPP and the rest of the applications.

- If you want to install any one or more of the individual programs, click the Custom button in the ISKM Installer and select the appropriate option. You can select more than one at a time by pressing ⌘ and clicking on them. Of course, if this is your first installation, installing individual programs won't provide all the pieces you need to access the Internet.

The most likely option is **Full MacPPP Install for EarthLink**, and if that's what you want to do, click the Install button. The ISKM Installer informs you that you must restart your computer after installing and asks if you would like to continue.

If you have any unsaved work open in other applications, click the No button and then the Quit button back in the Standard Install dialog. Save your work, quit the other applications, and repeat the steps to this point. When you are ready to install, click the Yes button when the ISKM Installer asks about restarting. Next, you're presented with a Standard File dialog that enables you to locate the ISKM3 Folder anywhere on your hard disk that you like.

Once you choose a location and click the Save button, the ISKM Installer proceeds to install everything, and when it's done, it informs you that everything has been installed correctly and forces you to click on a Restart button. Click it, and after your computer restarts, you're ready to configure the software that was installed for you.

Configuring MacPPP

First, you must configure MacPPP. The ISKM Installer tries to configure MacPPP for you, but there are some pieces of information that only you know. Open the Config PPP control panel that the ISKM Installer has installed for you in the Control Panels folder (see figure 1.2).

Note

To configure MacPPP, you must have an Internet account. If you have not made arrangements with an Internet provider, see the section "Connecting with Earthlink Network" later in this chapter for details on how to get an Internet account.

If you have your modem connected to your Printer port for some reason, select that port in the Port Name pop-up menu. If you use an Apple Express Modem or a Global Village PowerPort Mercury with either the PowerBook Duo or the PowerBook 500-series, choose Internal Modem from the pop-up menu. For most other internal PowerBook modems, the Modem choice is still correct. Then, from the PPP Server pop-up menu, select the modem configuration that most closely matches yours—if none look right, select

SupraFAXModem v.32bis, which uses a generic initialization string and should work with many Hayes-compatible modems.



Figure 1.2 Config PPP control panel.

Note

Other users contributed these modem configurations—I can't guarantee that they work because I don't have most of these modems to test on. You may have to use your modem manual or call your modem vendor to decide on a different modem initialization string. Also, I've only entered the most likely strings in Config PPP—there are even more in a file called *Modem Strings* in the MacPPP 2.0.1 folder.

Click the Config button to bring up the Server Configuration dialog (see figure 1.3).

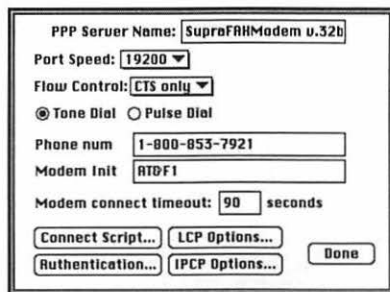


Figure 1.3 Config PPP Server Configuration dialog.

First, look at the phone number. It's an 800 number, which works fine from within the U.S. If you're dialing out through a large phone system in a company or in a hotel room, you may have to prefix the number with 8, or 9. In both cases, the comma tells the modem to pause for two seconds before dialing.

If you don't think the modem initialization string that's installed for you will work with your modem, enter the appropriate one in the Modem Init field. Check your modem manual—don't guess! Make sure your modem init string turns hardware handshaking on and XON/XOFF off.

If you're using a 28.8 Kbps modem and there wasn't a configuration for you in Config PPP's pop-up menu, choose 38400 from the Port Speed pop-up menu. If you're using a 14.4 Kbps modem, choose 19200. In both cases, if everything works right, you can increase that speed to 57600. Never select either 28800 or 14400—some modems simply don't work with those settings.

EarthLink Network TotalAccess uses PAP, or Password Authentication Protocol, so click the Authentication dialog and enter your userid and password (skip ahead to the section about calling EarthLink to get this information) in the dialog. Once you've entered your userid and password in the Authentication dialog (see figure 1.4), click the OK button to save your changes; and back in the Server Configuration dialog, click the Done button.

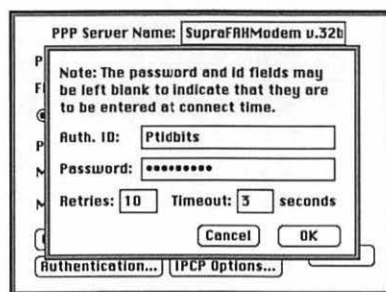


Figure 1.4 Config PPP Authentication dialog.

You should now be able to click the Open button in the Config PPP control panel to establish your connection to your PPP account. If you have troubles, read through chapter 11, "Making Connections," especially the "Troubleshooting Your Connection" section.

Note

Once you install all the software, configure MacPPP, and make the connection to your PPP account at EarthLink, that's when you run programs like Anarchie, Eudora, and NewsWatcher. MacPPP does nothing more than establish the connection.

The following four sections explain how to configure some popular Internet programs, some of which are necessary to use the CD properly. If you would like to learn more information about these programs, see chapter 12, "MacTCP-based Programs."

Configuring Eudora

To configure Eudora for use with your account, you must enter your POP account and real name in Eudora's Getting Started section in the Settings dialog box. See chapter 12, "MacTCP-based Programs," for instructions. EarthLink should have given you all of the information to enter into Eudora.

Configuring NewsWatcher

To properly use NewsWatcher or any of the newsreaders, you must know a few pieces of information. Most important is the name of your NNTP server (for EarthLink, it's **news.earthlink.net**), also called a news server, and the name of your SMTP, or mail, server (for EarthLink, it's **mail.earthlink.net**). If you use NewsWatcher, use Internet Config (see chapter 12, "MacTCP-based Programs") to enter this information so that other applications can use it later as well.

Configuring Anarchie

To use Anarchie, you must first enter your email address (which Anarchie uses as a password for anonymous FTP sites) in Internet Config. See chapter 12, "MacTCP-based Programs," for instructions on how to configure Internet Config and how to use Anarchie.

Configuring MacWeb

There are a few things you should know about the three sections of the CD (see following) that rely on MacWeb. First, MacWeb defaults to not showing images when you first install it. You can, of course, constantly choose Load Images from MacWeb's Options menu, but that's a pain. To turn on Autoload Images, choose Preferences from the File menu. Then, from the pop-up menu in the Preferences dialog, choose Format. Check the Autoload Images checkbox (see figure 1.5).

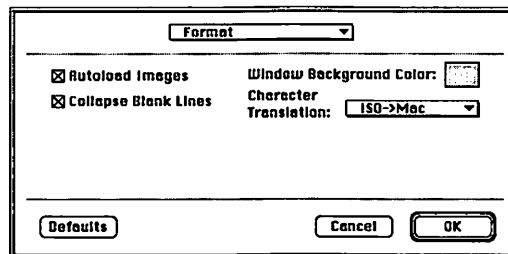


Figure 1.5 Set MacWeb to Autoload Images.

Second, MacWeb may have trouble displaying and working with certain images if you don't set a temporary directory on your hard disk. This problem occurs because MacWeb may run from the CD, and it can't save temporary files to the CD (since it's impossible to

write to the CD). So, while in the Preferences dialog, choose Files/Folders from the pop-up menu. Check the Default Folder for Temp Files button (see figure 1.6). As soon as you do that, MacWeb brings up a Standard File dialog for you to select a folder in which to store temporary files. Choose any folder on your hard disk that you like; I personally dump them in the System Folder or in the Preferences folder. When you're done, click OK to save your changes.

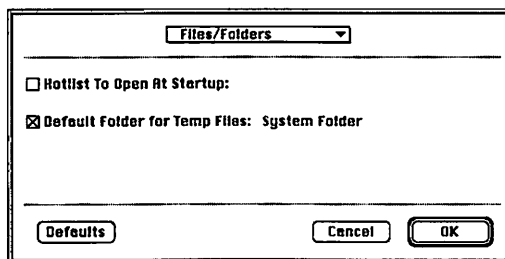


Figure 1.6 Set a default folder for MacWeb temp files.

Third and finally, some of the links in the Web pages are internal, that is, they refer to other files on the CD. Others are external, and require an active connection to the Internet. When you move the pointer over a link, look in the lower-left corner of the MacWeb window. If the URL that appears there starts with "file", the link is internal and you can follow it without needing an Internet connection. However, if the URL starts with anything else, such as "http", "gopher", "ftp", "news", or "mailto", then you know that the link is external, and if you don't have an Internet connection already in place, clicking on the link will make MacPPP try to dial your modem and connect. Keep this in mind while navigating around the CD, and you won't have to keep your Internet connection active unless you're actually browsing on the Web.

That's It

That's all there is to it—enjoy your Internet connection!

Installation Option Details

Installers are good at putting files in specific places, but they seldom tell you exactly where the various files have ended up. The following information explains where everything ends up on your hard disk, organized by installation option.

Full MacPPP Install for EarthLink

- **MacTCP:** MacTCP is installed in the Control Panels folder.



- **MacPPP:** Config PPP is installed in your Control Panels folder, PPP is installed in your Extensions folder, and PPP Preferences is installed in your Preferences folder. A MacPPP 2.0.1 folder containing documentation and modem strings is installed in your ISKM3 Folder.
- **QuickTime:** QuickTime is installed in your Extensions folder, and if you use a Power Macintosh, QuickTime PowerPlug is also installed in the Extensions folder.
- **Anarchie:** Anarchie is installed in your ISKM3 Folder. You should use Internet Config to configure Anarchie's Preferences with your email address.
- **Eudora:** Eudora is installed in your ISKM3 Folder. You must configure Eudora with your POP Account and real name before using it.
- **Internet Config:** Internet Config is installed in your ISKM3 Folder. You must configure your email address in Internet Config before you use Anarchie.
- **MacWeb:** MacWeb is installed in your ISKM3 Folder.
- **StuffIt Expander:** StuffIt Expander is installed in your ISKM3 Folder. You may wish to move the program to your desktop since you will probably be dropping many files on it to debinhex and expand them.
- **Get New Internet Programs:** This folder, containing the bookmarks that point at the latest versions of the most useful Internet programs stored on ftp.tidbits.com, is installed in your ISKM3 Folder.

MacTCP Only

- **MacTCP:** MacTCP is installed in the Control Panels folder. If you use this option to reinstall MacTCP after removing a corrupted version, make sure to restart after throwing out the old MacTCP, MacTCP DNR, and MacTCP Prep. Otherwise, your new version may retain the corruption.

MacPPP Only

- **MacPPP:** Config PPP is installed in your Control Panels folder; PPP is installed in your Extensions folder; and PPP Preferences is installed in your Preferences folder. A MacPPP 2.0.1 folder containing documentation is installed in the ISKM3 Folder that you save on your hard disk.

QuickTime Only

- **QuickTime:** QuickTime is installed in your Extensions folder, and if you use a Power Macintosh, QuickTime PowerPlug is also installed in the Extensions folder.

Anarchie Only

- **Anarchie:** Anarchie is installed in your ISKM3 Folder. You should use Internet Config to configure Anarchie's Preferences with your email address.

Eudora Only

- **Eudora:** Eudora is installed in your ISKM3 Folder. Before using it, you must customize Eudora with your POP Account and real name.

Internet Config Only

- **Internet Config:** Internet Config is installed in your ISKM3 Folder. You must configure your email address in Internet Config before you use Anarchie.

MacWeb Only

- **MacWeb:** MacWeb is installed in your ISKM3 Folder.

StuffIt Expander Only

- **StuffIt Expander:** StuffIt Expander is installed in your ISKM3 Folder. You may wish to move the program to your desktop for easy access.

Bookmarks for New Internet Programs Only

- **Get New Internet Programs:** This folder, containing the bookmarks that point at the latest versions of the most useful Internet programs stored on ftp.tidbits.com, is installed in your ISKM3 Folder.

Internet Interactive CD?

The *Internet Interactive CD* is pretty much self-explanatory once you click on the "Start Here" icon in the main CD window. However, I want to tell you about the main parts of the CD so you know what to look for.

What's on the CD?

This section is a QuickTime movie that introduces you to the various parts of the CD in a more graphical way than I can do in the text here.

Talk to Adam

Well, it's more like "Listen to Adam," but if you want to hear my dulcet tones and see my fuzzy visage, check out all the various movies in this section of the CD. After listening to the various clips, you can decide for yourself if I write like I talk, something I've been accused of in the past.

Step-by-Step

This collection of QuickTime movies show you how to configure and use the main Internet programs that are included on the CD. If you're the sort whose eyes glaze over at numbered lists of instructions, play these movies so you can get a sense of what you have to do. You can switch out of the Internet Interactive application to try the tasks you see in the movies.

Of course, even when you're able to start and stop a movie, it can be a little hard to follow along, so you may wish to configure the applications with help from both the movies, chapters 11, "Making Connections," and 12, "MacTCP-based Programs," and some of the chapters of *Internet Starter Kit for Macintosh* from the next section.

Read the Book

This book is only a few hundred pages, whereas the full *Internet Starter Kit for Macintosh, Third Edition*, weighs in at about 750 pages. We wanted to provide the full text, complete with screenshots, on the CD, and for that we chose the HTML format used by the World Wide Web.

When you click on the View button in the Read the Book section, the Internet Interactive application launches MacWeb, the Web browser that's installed for you (and also included on the CD in case you lose your copy on the hard disk). You can read the entire text of the book in MacWeb, and of course, you can click on any of the blue, underlined hyperlinks to move around in the book or even connect to Internet resources.

Best of Internet

The Best of Internet Web pages bring together lots of links to sites in several different categories, including Art, Entertainment, Government, Magazines, Film & Television, Commercial Sites, Computers, and Internet Indexes.

Of course, since these Web pages have been set in stone by appearing on the CD, the links to Yahoo and to the other Internet Indexes may prove to be the most useful in the long term. However, for getting a feel for the kind of things that are on the Web, this list will work well.

CD Art Tour

Next comes an Art Tour of some of the more impressive images available on the Internet. Small thumbnail images are followed by brief descriptions and links to the original sources on the Internet, which you can follow if you're connected to the Internet. Click on an image to view it in full size using JPEGView (or, if you've downloaded Netscape, you can view the image without JPEGView's help).

About Leonardo Multimedia

Last but not least, comes an About Leonardo Multimedia folder that contains the Leonardo ISKM Installer and information about the company and the people who created the *Internet Interactive CD-ROM*.

Connecting with EarthLink Network

We wanted to provide a single, simple method of connecting to the Internet from anywhere in the U.S., so we made a deal with an Internet access provider called EarthLink Network. It offers a number of plans, one of which should work well for you.

EarthLink Network TotalAccess USA 800

For a \$25 one-time setup fee, and \$24.95 per month, you get five hours of access via an 800 number (800-853-7921, but it's pre-configured into Config PPP for you as well). Each additional hour costs \$4.95, which is as cheap as you can find anywhere. EarthLink's TotalAccess USA 800 number is set up as the default phone number in Config PPP. This service will work well for everyone, and should also be a boon to travelers.

EarthLink Network TotalAccess Southern California

If you live in Southern California, you may be able to call a local number, avoiding the higher-priced 800 number service. This account costs \$25 for the one-time setup fee, and \$19.95 per month, which includes 50 primetime hours (defined as 9 AM to midnight, Monday through Friday) each month and unlimited non-primetime usage.

EarthLink Network TotalAccess USA

EarthLink has set up a national network to provide local access numbers in over 100 cities throughout the United States by September, 1995. The account has a \$25 setup fee and costs \$19.95 per month for 15 free hours. Additional hours are charged at \$1.95 per hour. When you call EarthLink to set up your account, it will send you, via fax or mail, instructions for using this network along with a list of phone numbers so you can see if there's a local one in your area. We anticipate that most people will be able to use this national network.

The main thing that will be different about using the national network is that it requires a modified form of your userid (probably ELN/userid). The instructions you get from EarthLink when you sign up will explain all the details.

Other Internet Access Providers

All of the software included on the CD will work with any Internet provider that offers PPP accounts. You may be able to find cheaper Internet access locally, and if you want to follow that route, I recommend that you read through all of chapters 16, 17, 18, and 20 in the version of *Internet Starter Kit for Macintosh* on the CD. Also check out appendix A and appendix B on the CD for some lists of Internet providers. Be aware that setting up the account may be a bit more confusing than working with EarthLink—we traded the flexibility of the way I'd done the installer in *Internet Starter Kit for Macintosh* for the simplicity of working with a single provider.

Again, if you're in Los Angeles, you might consider working with Leonardo Internet, since it will be essentially as easy to use it as EarthLink in LA.

Setting up an Account

The process of setting up an account with EarthLink is easy.

1. Call EarthLink at 213-644-9500 and tell the salesperson that you have the Internet Interactive CD and would like to set up a new account. Make sure to ask for information about the national network.
2. The salesperson will set up the account, print up a sheet of instructions with your userid, password, and other essential information, and then fax or mail it to you.
3. You pick the appropriate modem configuration in Config PPP, and enter your userid and password into the Authentication dialog (see the previous section on configuring MacPPP for details).
4. Once you've configured MacPPP, clicking the Open button or launching MacTCP-based program dials your modem and connects you to EarthLink and thus to the Internet. Make sure to close the connection with the Hard Close button in Config PPP when you're done to avoid accidental charges.

There's More...

But wait! Lest I sound too much like a late-night Ginsu knife commercial (has anyone *ever* bought one of those to cut beer cans?), I have in fact done even more.

ftp.tidbits.com

As I noted previously, I have bookmarks to the major Internet programs so you can download any one of them with merely a double-click of the mouse. But what if you want one of the less-popular programs, or a newer version than is available on the CD? Have I got an FTP site for you...

I've collected about 50 MB of software in the course of my testing, and the programmers just keep churning out newer and better versions. The only way to provide the latest version is via the Internet itself. So, I had a talk with the folks at Northwest Nexus, and they agreed to set up an FTP site for me that everyone on the Internet could access. This way, you will have a single site to visit for all of your Macintosh Internet applications and utilities.

The FTP site is called **ftp.tidbits.com**, and you can use any standard method of accessing an anonymous FTP site. Just use **anonymous** as your username and your email address as your password. If the machine rejects your full email address as a password, try using just your username and an @ sign, as in **ace@**; sometimes this particular FTP server is a bit finicky.

Here's a quick tutorial on connecting to the FTP site after you have installed and configured MacTCP, MacPPP, and Anarchie, by using Internet Config. Connect to the Internet. Launch Anarchie. If your Bookmarks window doesn't appear automatically, from the File menu choose List Bookmarks. Scroll down in the list until you see TidBITS, and then double-click on that bookmark and then go into the "tisk" folder (it's a hold-over from the first edition, which I referred to in shorthand as TISK, for The Internet Starter Kit). Alternatively, use the shortcuts I've set up for you with the bookmarks for ISKM HTML Programs, ISKM Internet Utility Programs, and ISKM MacTCP-based Programs that are higher up in the Bookmarks window.

When you double-click on one of these bookmarks, Anarchie connects to **ftp.tidbits.com**, switches into the proper directory, and lists the files. From there you can navigate around in the different folders by double-clicking on them. Double-clicking on a file retrieves it. It's that easy.

ISKM Home Page

The bookmarks and the FTP site are all fine and nice, but the more alert among my reading audience are no doubt asking themselves, "But how am I going to figure out that there's a new version of MacWeb, or Eudora, or whatever? And heck, how am I going to find anything I want on the Internet anyway?" Good question, alert readers, and the simple answer is that you'll use MacWeb or another Web browser to connect to the ISKM Home Page at:

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

I've set the copy of MacWeb that's installed for you to connect to this page by default. However, if you've set a different home page in MacWeb previously, or wish to use Netscape or another Web browser, you can still use the URL for the ISKM Home Page as your default home page. All the programs have various ways, usually located in the Preferences, of setting a default home page.

The ISKM Home Page has a number of links to the most important sites on the Internet, catalogs of resources like Yahoo, search engines like WebCrawler, important FTP sites like Info-Mac and UMich, and even a few of the major companies in the Macintosh industry, such as Apple and Microsoft. These links should be your starting point for any exploration of the Internet, and rest assured that if I find additional sites that I consider equally as useful, I'll add them.

In addition, at the top of the ISKM Home Page is a link, called "ISKM Macintosh Internet Software Updates," to another Web page that continually tracks the latest versions of all the main Internet applications for the Macintosh (so stop in every now and then and see what's changed). You can use your Anarchie bookmarks to retrieve any of these programs, or you can use a Web browser (although Anarchie is better at retrieving files).

Again, these pages are a public Internet resource and anyone is welcome to use them. If you find them useful, I'm glad, and I hope they might inspire you to think what you could contribute to the net someday.

On to the Small Talk

Now that you've installed the software, we should introduce ourselves, and let you get a feel for what the Internet is, what it's good for, and where it came from. That's what the next few chapters cover, so read on!

Chapter 2



Welcome

Welcome to *Simply Amazing Internet for Macintosh*. I have two goals for this book, at least one of which hopefully applies to you as either its prospective buyer or proud new owner. First, I want to tell you about the Internet—what it is and why it's so wonderful (and I mean that in all senses of the word, especially the bit about becoming filled with wonderment)—and introduce you to a number of the services and resources that make it one of humankind's greatest achievements. Second, I want to show you how to gain access to the Internet using the tools included on the accompanying CD-ROM.

The pages that lie before you will not tell you absolutely everything that I know about the Internet because this book is based on my *Internet Starter Kit for Macintosh, Third Edition*, and that book is two to three times larger than this book. However, thanks to the capaciousness of the CD-ROM, we were able to include the full text, with screenshots, of *Internet Starter Kit for Macintosh* in a format that enables you to read it onscreen.

But before I start, let's skip the small talk and introduce ourselves.

Who Are You?

I haven't the foggiest idea who you are. That's not true, actually; I can make a couple of guesses. You probably are a Macintosh user, because if you aren't, only about half of this book will hold your interest and the CD won't work. You probably are also interested in the Internet; otherwise, you're wasting your time entirely. Given those minor prerequisites,

Simply Amazing Internet for Macintosh should provide hours of educational entertainment, just like Uncle Milton's Ant Farm. The major difference is that the Internet Ant Farm is worlds bigger than Uncle Milton's, and if you go away on vacation, all the Internet ants won't keel over—although you may be tempted to do so when you get back and see how much you have to catch up on. The Internet never stops.

I've written this book for the individual, the person behind that most personal of personal computers, the Macintosh. In the process, I undoubtedly will disappoint the die-hard Unix system administrators and network gurus who talk about X.400 and TCP/IP in their sleep (which doesn't come often because of the amount of Jolt cola they consume). No, this book and its CD are for ordinary people who have a Mac with a CD player, a modem, and the desire to start using the Internet. If you fit that description, you're all set.

What Do You Need?

This book, of course—why do you think I wrote it? But beyond that...

First, you need a Macintosh with at least 8 MB of RAM, at least a 13-inch color monitor (color is required, and the screen must display at least 640 by 480 pixels) and a CD player. You don't need a fast Macintosh, although it would be nice. When I wrote the first edition of this book, I used an elderly SE/30 with an external color monitor and CD player, and although I've since moved up to a Centris 660AV, most things I do on the Internet haven't changed much with the faster Mac.

Second, you need some type of physical connection to the Internet. This connection may take the form of a local area network at work or, more likely, a modem. A 2,400 bps modem works, though only barely. The faster the modem, the better. A 28,800 bps modem is the best modem you can have. If you start out with a 2,400 bps modem, be prepared to buy a new one soon. That speed will become intolerable quickly, and why make something as fascinating as the Internet intolerable?

Third, I recommend that you use System 7 or later, if only because I haven't used System 6 in over four years and have no idea whether the software included on the CD works under System 6. That's not entirely true—some of the basic programs will work—but many others, including the best ones, now require System 7. All of my instructions assume that you are using System 7. If you need to upgrade, talk to your dealer.

Fourth, you need an account on a host machine somewhere. Luckily for you, we've worked out a deal for you with a provider, EarthLink Network, based in Los Angeles, California. EarthLink offers local dialup access for people who live near LA, access via a national network of phone numbers for about \$2 per hour, and finally 800 number access (at about \$5 per hour) for people on the road or in areas that don't have a local access number. Those of you in Los Angeles might also consider Leonardo Internet, another Internet provider in Los Angeles that's run by the same people who created the CD-ROM.

Fifth, you need a certain level of computer experience. This stuff simply is not for the Macintosh novice. If you don't know the difference between a menu and a window, or haven't figured out how to tell applications and documents apart, I recommend that you visit your local user group and ask a lot of questions. A number of excellent books also are available. My favorite is *The Little Mac Book*, by Robin Williams. You can also read your manuals, but frankly, as good as Apple's documentation is, Robin does a better job.

Finally, you may need to adjust your expectations. The Internet is not a commercial service like America Online or CompuServe. Customer service representatives are not available via a toll-free call 24 hours a day. The majority of people on the Internet have taught themselves enough to get on or have been shown just enough by friends to connect to the Internet. The Internet is very much a learning experience; even with the information and guidance that I provide, there's simply no way to anticipate every question that might come up through those first few days. The Internet is what you make it—so don't be shy. No one greets you on your first dip in, but at the same time, people on the Internet are some of the most helpful I've ever had the pleasure to know. If you are struggling, just ask and someone almost always comes to your aid. I wish that were true outside of the Internet as well.

Who Am I?

"Who am I?" is a question that I often ask myself. In the interests of leaving my autobiography for later, I must limit the answer to the parts that are relevant to this book. My name, as you probably figured out from the cover, is Adam Engst. I started using computers in grade school and had my first experience with a mainframe and a network while playing Adventure over a 300 baud acoustic modem (you know, where you dial the number and stuff the receiver into the modem's rubber ears) on a computer that my uncle used in New York City. I used microcomputers throughout high school, but upon entering Cornell University I learned to use their mainframes. In my sophomore year, I finally found the gateway to BITNET (the "Because It's Time" Network) in some information another user had left behind in a public computer room. Finding that initial bit of gateway information was like finding a clue in Adventure—but don't worry, it's not that difficult any more. From BITNET I graduated to using a computer connected to Usenet (the User's Network, generally synonymous with "news"), and around the same time I learned about the vast Internet, on whose fringes I'd been playing.

After graduating from Cornell in 1989, I set up my own Internet access using QuickMail for Macintosh. QuickMail was overkill for a single person because it's designed to be a network electronic mail program, so I eventually switched to a more appropriate program called UUCP/Connect. Several years ago, my wife and I moved from Ithaca, New York (where we had grown up and where Cornell is located) to the Seattle, Washington area. In the process, I learned more about finding public-access Internet hosts in a place where you know no one in person. In many ways, the Internet kept me sane those first few

months. My Internet access changed over time, and although I kept using UUCP for email, I switched first to a SLIP account with Northwest Nexus before writing the first edition of *Internet Starter Kit for Macintosh* and then, for the second edition, to a PPP account. Several months after the second edition of *Internet Starter Kit for Macintosh* came out, I finally broke down and got a direct connection to the Internet. My Macs are permanently on the Internet now.

Throughout this Internet odyssey of the last nine years, I've used the nets for fun, socializing, and general elucidation. In the last five years, I've also written and edited a free, weekly, electronic newsletter called *TidBITS*. It focuses on two of my favorite subjects: the Macintosh and electronic communications. *TidBITS* is both a product and a citizen of the Internet. It has grown from a 300-person mailing list that once crashed a Navy computer running old mail software, to an electronic behemoth that lives on every network I can find—boasting an estimated 125,000 readers in over 55 countries.

So that is the reason I'm writing this book (well, there are those incriminating photographs of publishing industry VIPs that I have digitized and poised to distribute to the net at large). Any questions? I hope so, but hold off until you've finished the book. And for those of you already marking things up with those nasty yellow highlighters, don't; I promise there is no quiz awaiting you.

Changes

Keep in mind that the Internet changes quickly and constantly; trying to capture it in a snapshot requires high-speed film. I've got that film, so the image of the Internet that I present here isn't blurry or out of focus, but it's impossible to cover, or even discover, everything that deserves mention. Thus, if you want to keep up with the changes, it's partly up to you to get out on the Internet and see what's happening. I can help a bit, and I do, with some World Wide Web pages that track the latest Macintosh Internet programs and other events of importance, but in the end, you have to decide the extent to which you want to stay up to date.

The various books I've written for Hayden Books have distinctly changed my life, for the better. I've been asked to speak at conferences, interviewed via email and on the radio, filmed for TV, and on interminable cross-country airplane flights I was fed food that was pre-chewed for my convenience by weasels. But the reason I put up with all the hassle is that I truly love the Internet and believe that it's worth preserving, protecting, building, and explaining. If I can infect others with my enthusiasm for the Internet, I think the world becomes a better place. And that's the goal in the end.

Chapter 3



Why Is the Internet Neat?

Unless you've just idly picked up this book based on its bright cover while waiting for your spouse to choose the right gift for Aunt Millie's birthday, you probably have some sense that you should be interested in the Internet. Given the Clinton administration's emphasis on a national data highway system, many a poor reporter has written or broadcast a story on this Internet thing.

Those stories almost always make those of us who live and breathe the Internet cringe because they almost always miss the point. The stories either crow about the technological achievement and vast worldwide coverage of the Internet (while failing to explain that it is definitely not a commercial service staffed by friendly nerds in white coats, and ignoring its human dimension), or they provide a gratuitous human interest story about how two people met on the Internet and got married eleven days later because typing to each other was such a moving experience. Sure, this stuff happens, but such gee-whiz stories never touch on the commonplace parts of the Internet: the discussion groups, information databases, and selfless volunteer work that keeps the whole thing running. That's a shame, and I vow to avoid that slippery slope.

But I should be talking about why you should be interested in the Internet, instead of ragging on the mediocre descriptions from people who apparently aren't. Keep in mind that I may miss your favorite reason to use the Internet—

one woman's Brownian motion generator is another man's cup of tea. In addition, remember that technology is seldom used for its intended purpose. The Internet started as a method of linking defense researchers around the country; it has grown beyond that use in ways its creators never could have imagined.

Electronic Mail

For many people, electronic mail (or email) is the primary reason to get on the Internet; they simply want to be able to send mail to someone else on the Internet. Once you're on, though, you're likely to strike up many new friendships and end up with a long list of electronic correspondents. Because it's quick and easy, email is an excellent way to stay in touch, even with people whom you regularly talk to on the phone. Even though I talk to my parents often, I also send them email because it's more appropriate for quick notes. Email messages are even better than an answering machine for conveying simple information. At one point, for example, the local Macintosh user group held steering committee meetings at my house. I could have called all the steering committee members before each meeting to remind them about it, but because all I wanted to say was, "Don't forget the meeting tomorrow night," contacting them was easiest via email.

Email sometimes gains the least likely converts. One friend of mine is best described as a telephobe—he hates talking on the telephone and only has one at his house, out of necessity. He had been equally disparaging of computers and email until he was forced to try it, after which he became an instant email proponent. He discovered that with email, he no longer had to play telephone tag with coworkers or try to arrange meetings to talk about simple topics. Email enabled him to work more flexible hours because he didn't care when his coworkers were present, and their email was waiting whenever he wanted to read it.

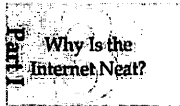
Discussion Groups

A large number of people read and participate in the thirty or so discussion groups, also called *newsgroups*, about the Macintosh, and far more people contribute to thousands of other non-technical discussions. Several years ago, when I went away on a bike trip, my Macintosh started sounding the Chords of Death and displaying the sad Mac face along with an error code. My wife couldn't contact me to tell me about it, but she posted a help message on one of the Macintosh discussion groups. Within a few days she had received answers from Macintosh experts around the world, all telling her that code meant we had a bad memory card. (Luckily, the card turned out to be only badly seated.)

Similarly, when we were in the process of buying a car, I started reading appropriate messages on one of the discussion groups dedicated to talking about cars. The messages were of some help, but I wish I had known then that there was an entire discussion group devoted to Hondas, the make we were looking at most.

Software

For many Macintosh users, some of the most immediately useful and interesting things about the Internet are the *file sites*. File sites are computers on the Internet that are accessible to everyone (more or less) and store thousands of the latest and greatest *freeware* and *shareware* (where you pay the author if you use the program) programs for the Macintosh. An equal or greater number of file sites exist for other platforms, most notably for the omnipresent PCs from IBM, Compaq, Dell, and the other 17 million clone makers. Finding specific numbers is difficult, but I think it's safe to say that thousands of people download files every day from the most popular *archive sites* (just another name for file site).



Information at Your Electronic Fingertips

The popularity of email and newsgroups notwithstanding, the massive databases of information impress some people the most. A couple of years ago, a friend came over to look at a QuickTime movie of the Knowledge Navigator film clip. The Knowledge Navigator is ex-Apple CEO John Sculley's idea of what information access will be like in the future—an anthropomorphic "talking head" that acts as an *information agent*, searching through massive databases of information on the user's command. The Knowledge Navigator film portrays a professor preparing for a class discussion about deforestation in the Amazon rain forest by looking at data retrieved by his electronic agent.

The film is fairly neat, but after watching it, I remembered that I also wanted to show my friend Wide Area Information Server (WAIS). Using the Macintosh WAIS software, we connected to WAIS and typed in our query, "Tell me about deforestation in the Amazon rainforest." After about 10 seconds, WAIS returned a list of 15 articles from various sources that dealt with just that topic, sorted by relevance. Talk about knocking someone's socks off—my friend was staring, mouth open, tongue lolling, and completely barefoot, so to speak. Although WAIS doesn't have an infinite number of databases, it does have over 700 (including *TidBITS*), and more appear all the time.

Although I have no numbers to back this up, I get the impression that the largest quantity of raw information is available via FTP (File Transfer Protocol). Even if that's not true, more data is still transferred via FTP than any other method on the Internet, although the World Wide Web is catching up fast. The freeware and shareware programs for the Mac that I mentioned above are available via FTP, as are electronic editions of books, newsletters such as *TidBITS*, fiction magazines such as *InterText*, and huge numbers of other files.

Gopher, another method of transferring information over the Internet, is also quite popular. It's easy to set up a Gopher server—so anyone who has good information and a dedicated Internet connection can do it. WAIS databases require high-powered computers, whereas a ten dollar shareware program from an Australian programmer named Peter Lewis enables someone to set up a Gopher server on a Macintosh (I've done it for

some friends here in Seattle—it took about an hour). More than 2,300 Gopher servers exist today, and the information available on them includes things like Macintosh price lists at major universities (they often contain Apple's prices as well, making them useful for comparison even if you don't attend a university), Internet statistics, tech support information from Apple, and press releases from the U.S. government.

Finally, a vast amount of information appears on a daily basis on the World Wide Web, a service created by CERN, the high-energy physics research lab in Switzerland. Other methods of providing information over the Internet have been pretty much restricted to text until the data is downloaded to a Mac or PC. But the World Wide Web supports text with fonts, sizes, and styles, graphics within the text, sounds, animations, and movies; and all of it is interconnected with hypertext links. For many folks with information to provide to the Internet, the World Wide Web is the way to go. For instance, I've seen Web pages ranging from a beautiful collection of fractals (some even animated), to a wonderful museum-style paleontology exhibit at the University of California at Berkeley, to an extensive set of pages from a group called INFACt Online devoted to a campaign to stop tobacco companies from marketing cigarettes to children.

The Lemming Factor

Aside from the personal communications, the discussions on every imaginable subject (and many you'd never imagine), and the databases of information, the Internet is neat for yet another reason: It's what I sometimes call the "lemming factor." That is, if so many people from so many cultures and walks of life are connecting to the Internet, something has got to be there. Don't scoff; no one makes all these people log on every day and spend time reading discussion lists and sending email. People aren't forced to increase Internet traffic at a whopping rate of 20 percent per month. They use the Internet because they want to, and few people are happy when they lose Internet access for any reason. And as much as "lemming factor" may imply people are getting on Internet because their friends are, they aren't doing it from peer pressure (well, okay, so I hassled my parents into getting connected, but they love it now). People connect to the Internet because it is becoming more than just an elite club of technoweenies—it has become a virtual community in and of itself.

The allure of the Internet sets it apart from other communities such as religious, charitable, or humanitarian groups. No implied theological punishment exists for avoiding the Internet, and although its attraction somewhat resembles that of volunteer groups such as the Red Cross, those organizations often depend on people's belief systems. The Internet continues to thrive because of the volunteer labor pumped into it; but also important is the fact that it provides as much information as an individual can handle, and in this day and age, information is power.

The Internet Is What You Make of It

Whatever advantage you want to take of the Internet, remember two things. First, the information available on the Internet has generally avoided the processing introduced by the mass media (although more of the mass media arrives on the Internet constantly as well). If you want some unfiltered opinions on both sides of any issue ranging from the death penalty to abortion to local taxes, people usually are discussing the issue at length somewhere on the net. Because of the lack of filtering, you may read a bit more about any one subject than you do in the mass media.

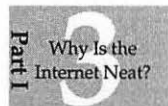
Second, you get only the information you want. For about a year, my wife and I followed a weekly routine with the Sunday *Seattle Times*. First, we'd compete for the comics and then for the *Pacific Magazine*, which has in-depth articles. Then we'd settle down: I'd read the Sports section and the Business section, and my wife proceeded to the Home & Garden section. Good little stereotypes, weren't we? The point is that I was completely uninterested in reading at least three-quarters of the two-inch thick stack of paper, and so was my wife. So why were we paying for the entire thing only to bring it home and recycle half? A good question, and one that newspaper publishers should get their duffs in gear and answer.

Tonya and I answered it by ceasing to bother with the Sunday paper. Not only was it a waste of paper resources, especially considering that we didn't read most of it, but it was a waste of time to flip through much of the parts that we did read. Instead, I've started getting the news I want on the Internet, through a combination of mailing lists, newsgroups, and Web pages that cover my interests closely. I can't get all the comics that I'd like to read yet, but *Dilbert* from Scott Adams, an Internet-only cartoon called *Dr. Fun*, and some of the *Slugs!* cartoons that my friend Dominic White drew for *Internet Explorer Kit for Macintosh* have all appeared in recent months.

The same overkill problem applies to junk mail. I instantly throw out about 90 percent of the *snail mail* (the Internet term for paper mail) I get, whereas almost all email I get is at least worth reading.

On the Internet, when all is said and done, I get only what I ask for. Periodically, my interests change, so I switch things around, but I don't have to read, or even deal with, topics that either bore or irritate me—such as anything unpleasant happening in Northern Ireland or Beirut. Try avoiding such topics in the mass media. It's just not possible.

Now that I think of it, there's a third point I want to make about information on the Internet. Most of it, as I said, is free of media processing. That's because most of the information comes from individuals and small groups rather than large publishing conglomerates that own hundreds of newspapers and magazines around the world. Even though I'm not going to tell you anything about how to set up an Internet machine to provide information over the Internet, be aware that you as an individual don't necessarily need



your own machine. You could run a small mailing list from a Mac, and you could easily post a newsletter or report of some sort to discussion lists without a dedicated machine. And, if what you want to do requires an FTP site or mainframe that can run mailing list software, ask around; someone may be willing to provide that sort of access to you. This is how I've published *TidBITS* for the last five years. As long as you're providing useful information for free, you'd be surprised how many people may step forward to help you.

Champing at the Bit

I know you're all excited about the Internet now that you know why it is so neat. But, you're probably saying to yourself, "Self, it sure sounds like I can do lots of cool things on the Internet, but just what the heck is this Internet thing, anyway?" Glad you asked yourself that question because that's precisely what we will talk about next.

Chapter 4



What Is the Internet?

What is the Internet? That question is tremendously difficult to answer because the Internet is so many things to so many different people. Nonetheless, you need a short answer to give your mother when she asks, so here goes:

The Internet consists of a mind-bogglingly huge number of participants, connected machines, software programs, and a massive quantity of information, spread all around the world.

Now, let's see if I can put those various parts into some kind of meaningful context.

Size

To say the Internet is big—in terms of people, machines, information, and geographic area included—is to put it mildly. How big is it? Let's take a look and see.

People

The Seattle Kingdome seats approximately 60,000 people for a sellout Mariners baseball game (a once-in-a-lifetime experience for the hapless Mariners). That's about the same number of people who read a single, mildly popular newsgroup on the Internet. If all of the estimated 30 million people on the Internet (according to some sources; others estimate much lower numbers) were to get together, they'd need 500 stadiums each the size of the Kingdome to have a party. I could calculate how many times that number of people would reach to the moon and back if we stacked them one on top of another, but I think I've made my point.

Machines

In the infancy of the computer industry, IBM once decided that it did not need to get into the computer business because the entire world needed only six computers. Talk about a miscalculation! Many millions of computers of all sizes, shapes, and colors have been sold in the decades since IBM's incorrect assumption. An estimated 4.9 million of these (4,852,000 as of January 1995, for those of you who like the digits) are currently connected to the Internet. I keep having trouble with these numbers because they change so frequently. In the first edition of *Internet Starter Kit for Macintosh*, I used 1.7 million computers as the basic number. I had to change it to that at the last minute because the manuscript I'd originally sent to Hayden Books used 1.3 million, the number from a few months before. When we published *Internet Starter Kit for Macintosh, Second Edition* in August of 1994, I updated the number to 2.3 million, and here I am, not even a year later, updating it to 4.9 million. See Table 3.1, which lays out the data collected by Network Wizards over a number of years. You can find the latest version of this data via the Web at:

<http://www.nw.com/>

Table 3.1
Internet Host Growth

<i>Date</i>	<i>Number of Hosts</i>
January '95	4,852,000
October '94	3,864,000
July '94	3,212,000
January '94	2,217,000
October '93	2,056,000

Date	Number of Hosts
July '93	1,776,000
April '93	1,486,000
January '93	1,313,000

Information

I can't pretend that the Internet offers more pieces of useful information than a good university library system, but that's only because a university has, in theory, a paid staff and funding for acquisitions and development. Information on the Internet is indeed vast, but finding your way around proves a daunting task. However, neither could I pretend that finding a given piece of information in a large research library would be any easier without the help of a skilled reference librarian.

Information on the Internet also changes and seems to appear more quickly than in a physical library, so you never know what's arrived since you last visited. Also, keep in mind that Internet information is more personal and fluid than the sort of information in a library. Although you may not be able to look up something in a reference work on the Internet, you can get ten personal responses (some useful, some not) to almost any query you pose.

Geographic Size

Explaining how large the Internet is geographically is difficult because, in many ways, messages traveling over the network connections don't give a hoot where they are going physically. Almost every industrialized nation has at least one machine on the Internet, and more countries come online all the time. But geographical distance means little on the net. For example, I mail issues of *TidBITS* to our mailing list on Monday night. People down the road from me find it in their mailboxes on Tuesday morning, as do subscribers in New Zealand and Norway.

Note

Although I don't know if this is still true, several years ago Norway had the highest per-capita density of Internet machines.

A friend described the Internet as ranging from Antarctica to the space shuttles, from submarines to battle tanks, from a guy riding a bicycle around the globe to others crossing oceans in a yacht, from kids in kindergarten to the most eclectic gathering of brains.... Well, you get the idea.

Perhaps the best way of wrapping your mind around the Internet is to recall the old joke about blind men all giving their impressions of an elephant based on what they can feel. Like that elephant, the Internet is too large to understand in one mental gulp (see figure 3.1).

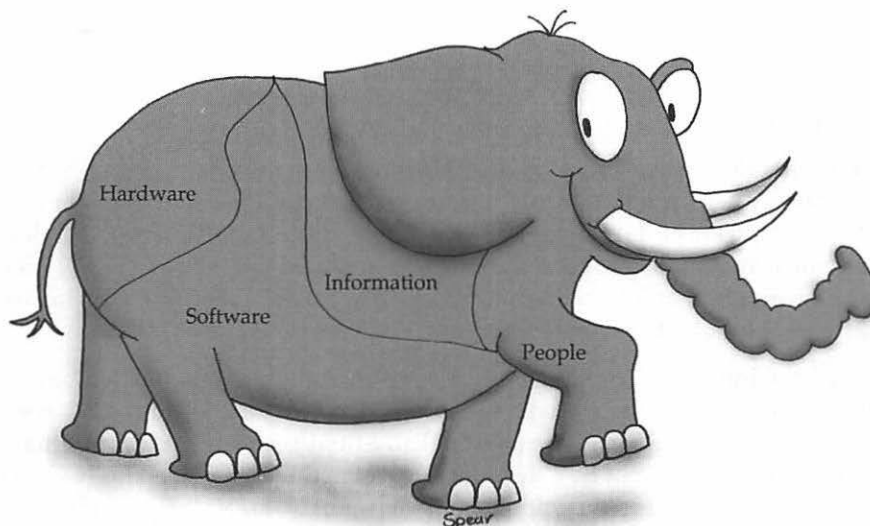


Figure 3.1 The Internet Elephant—Elephantidae internetus.

Some people may think of the Internet in terms of the people that are on the net (this is my favorite way of looking at it). Technical people may insist that the machines and the networks that comprise the physical Internet are the crux of the matter. Software programmers may chime in that none of it works without software. Others may feel that the essence of the Internet lies in the information present on it.

In fact, all of these people are equally right, although as I said, I personally prefer to think of the Internet as millions of people constantly communicating about every topic under the sun. The amount and type of information, the hardware and software will all change, but the simple fact of people communicating will always exist on the Internet.

People—Doing What They Do Best

The most important part of the Internet is the collection of many millions of people, *homo sapiens*, all doing what people do best. No, I don't mean reproducing, I mean communicating.

Communication is central to the human psyche; we are always reaching out to other people, trying to understand them and trying to get them to understand us. As a species, we can't shut up. But that's good! Only by communicating can we ever hope to solve the problems that face the world today. The United Nations can bring together one or two representatives of each nation and sit them down with simultaneous translations. But via the wire and satellite transmissions of the Internet, anyone can talk to anyone else on the Internet at any time—no matter where they live.

I regularly correspond with friends (most of whom I've never met) in England, Ireland, France, Italy, Sweden, Denmark, Turkey, Russia, Japan, New Zealand, Australia, Singapore, Taiwan, Canada, and a guy who lives about 15 miles from my house. (Actually, my wife and I finally broke down and went to visit the guy up the road, and he's since become one of our best friends in Seattle and the managing editor of *TidBITS*.) I've worked on text formatting issues over the networks with my friend in Sweden, helped design and test a freely distributable program written by my friend in Turkey, and co-written software reviews with a friend in New Zealand, who had been one of my Classics professors back at Cornell. On the net, where everything comes down to the least common denominator of *ASCII* text, you don't worry about where your correspondents live. Although people use many languages on the net, English is the *de facto* language of the computer industry, and far more people in the world know English than English speakers know other languages.

During the Gulf War, while people in the U.S. were glued to their television sets watching the devastation, people in Israel were sending reports to the net. Some of these described the terror of air raid sirens and worrying about SCUD missiles launched from Iraq. No television shot of a family getting into their gas masks with an obligatory sound bite can compare with the lengthy and tortured accounts of daily life that came from the Israeli net community.

The Internet also helped disseminate information about the attempted coup in the former Soviet Union that led to its breakup. One Internet friend of mine, Vladimir Butenko, spent the nights during the events near the Parliament. When everything seemed to be clear, he went to his office, wrote a message about what he'd seen, and sent it to the Internet. His message was widely distributed at the time and even partially reprinted in the *San Jose Mercury News*.

Although people on the Internet are sometimes argumentative and contentious, is that entirely bad? Let's face it, not all the events in the world are nice, and people often disagree, sometimes violently. In the real world, people may repress their feelings to avoid conflict, and repression isn't good. Or people may end up at the other extreme where the disagreement results in physical violence. On the Internet, no matter what the argument (be it about religion, racism, abortion, the death penalty, the role of police in society, or whatever else), there are only three ways for it to end. First, and mostly likely, all parties involved may simply stop arguing through exhaustion. Second, both sides may

agree to disagree (although this usually only happens in arguments where both sides are being rational about the issues at hand). Third, one person may actually convince another that he is wrong; though I doubt this happens all that often since people hate to admit they're wrong. But notice, in none of these possibilities is someone punched, knifed, or shot. As vitriolic as many of these arguments, or flame wars, can be, there's simply no way to compare them to the suffering that happens when people are unable to settle their differences without resorting to violence.

Most of the time on the net, an incredible sense of community and sharing transcends all physical and geopolitical boundaries. How can we attempt to understand events in other parts of the world when we, as regular citizens, have absolutely no clue what the regular citizens in those other countries think or feel? And what about the simple facts of life such as taxes and government services? Sure, the newspapers print those info-graphics comparing our country's tax burden to that in other countries. But this information doesn't have the same effect as listening to someone work out how much some object, say a Macintosh, costs in France once you take into account the exchange rate and add an 18 percent VAT (value-added tax), which of course comes on top of France's already high income taxes. It makes you think.

If nothing else, that's the tag line I want to convey about the Internet. It makes you think.

Maybe with a little thought and communication, we can avoid some of the violent and destructive conflicts that have marked world affairs. Many of the Internet resources stand as testament to the fact that people *can* work together with no reward other than the satisfaction of making something good and useful. If we can translate more of that sense of volunteerism and community spirit back into the real world, we stand a much better chance of surviving ourselves.

Hardware

Getting down to the technical data, more than 4.9 million computers of all sizes, shapes, and colors make up the hardware part of the Internet. In addition to the computers are various types of network links, ranging from super-fast T-1 and T-3 lines all the way down to slow 2,400 bps modems. T-3 was also the code name for Microsoft Word 6.0—but I digress. That often happens when I'm talking about relatively boring things like networks, because what's important is that the Internet works. Just as with your telephone, you rarely notice its technical side unless something goes wrong.

Computers

The computers that form the Internet range from the most powerful supercomputers from Cray and IBM all the way down to your friendly local Macintosh and garden-variety PC clones. You can split these machines into two basic types: *host* computers and *client*

computers. Host computers are generally the more powerful of the two, and they usually have more disk space and faster connections. Although, I don't want to imply that host machines must be fancy, expensive computers; Apple's popular FTP site at one point ran on an aging Mac II (however, it did use Apple's implementation of Unix, called A/UX).

Similarly, client machines also can be large, powerful workstations from companies like Sun and Hewlett-Packard. Because their task of sending and receiving information for a single person (as opposed to many people) is more limited, clients generally require less processor power and storage space than host machines. Basically, it can be a waste to use a \$10,000 Unix workstation as an Internet client (although they can make great client machines, if you have the money to throw around). In my opinion, microcomputers—Macs in particular—make the best clients. Why spend lots of money and a large amount of configuration time when an inexpensive, simple-to-set-up Macintosh does the job as well or better?

For the most part, I look only at client hardware and software in this book. The gritty details of setting up an Internet host and the many programs that run it aren't all that interesting to most people, not to mention the fact that I haven't the foggiest idea of how to configure a Unix workstation to be an Internet host. I'll leave those tasks to the wonderful people who are already doing them. (First rule of the Internet: Be extremely nice to your system administrator.) If you want to get into the administration end of things, O'Reilly & Associates publishes a long line of books on Unix and network administration, including one called *Managing Internet Information Services* that is a must-read for anyone who wants to run Internet servers on a Unix machine.

Note

You don't need to use Unix on an Internet host, and in fact, you can run an FTP, Gopher, or World Wide Web server on a Mac with no trouble. But for practical reasons this setup requires a fast and constant connection to the Internet, and again, I'm aiming this book more at users of Internet information, not information provider wanna-bes. Maybe the next book.

Networks

In basic terms, two computers attached together form a *local area network*, and as that network grows, it may become connected to other independent local area networks. That configuration is called an *internet*, with a small *i*. The Internet, with a capital *I*, is the largest possible collection of inter-connected networks. I could spill the gory details of what networks are connected to the Internet and whether they are true parts of the Internet (as defined by using a set of *protocols* called *TCP/IP*, or Transmission Control Protocol/Internet Protocol, the language that Internet machines speak), but that information wouldn't be very useful to you.

Note

An Internet old-timer once commented that “Internet” technically only applies to machines using TCP/IP protocols. He said the term once proposed for the collection of all the interconnected networks, no matter what protocols they used, was “WorldNet.” That term seems to have faded into obscurity—unfortunately, since it’s rather apt. Common usage now includes “Internet” (the safest term, though it’s technically inaccurate), “the net” (sometimes capitalized), and “cyberspace” (a heavily overused term from William Gibson’s science-fiction novel *Neuromancer*, which, ironically enough, he wrote on a manual typewriter with images of video games, not the Internet, in his head). Lately, the term “information superhighway” (an unfortunate term that has spawned imagery of toll booths, speed bumps, on-ramps, and road kill, but which means almost nothing in the context in which it’s generally used) is in vogue.

Worrying about the specific network protocol details is generally pointless these days because many machines speak multiple languages and exist on multiple networks. For instance, my host machine speaks both TCP/IP as an Internet client and UUCP (Unix to Unix CoPy) as a UUCP host. My old machine at Cornell existed in both the Internet and in the BITNET worlds. The distinctions are technical and relatively meaningless to the end user.

Modems and Phone Lines

For most people using a microcomputer such as a Macintosh, a *modem* generally makes the necessary link to the Internet. Modem stands for modulator-demodulator (glad you asked?), and it enables your computer to monopolize your phone, much like a teenager. You may not need a modem if you study or work at an institution that has its local area networks attached to the Internet. If you are at one such site, count yourself lucky and ignore the parts of this book that talk about finding connections and using the modem. But remember those sections exist; one day you may leave those connections behind, and nothing is more pitiful than someone pleading on the nets for information on how to stay connected after graduation or other significant life changes.

Certain new types of connections, including high-speed ISDN (Integrated Services Digital Network) may be the death of the modem as we know it. However, even ISDN connections require a box called a terminal adapter to enable the computer to appropriately pass data over the ISDN lines. In addition, even for folks with normal telephone line connections to the Internet, the modem itself may fade into the background—or rather, into the innards of the computer. It’s already possible to almost completely emulate a modem in software on Apple’s Macintosh 660AV, 840AV, and Power Macintosh computers. Eventually, wireless modems may become common, so the details of making a connection may fade away entirely. Or at least that’s what I hope happens.

Note

In an exaggerated show of acronym making, normal phone service is known as POTS, or Plain Old Telephone Service. Don't the people who came up with this have anything better to do with their time?

It's beyond the scope of this book to tell you what sort of modem to buy, but most reputable modem manufacturers make fine modems with long or lifetime warranties. Some companies sell extremely cheap modems, which often work fine in most cases, but you may also get what you pay for.

Note

I don't want to bash specific manufacturers, but many Macintosh users have had good luck with modems from Supra and Global Village, and I personally have owned modems from these two companies along with modems from Telebit and Practical Peripherals. Apple's Express Modem can be a bit flaky at times, but it's one of only two choices for the PowerBook Duos. (Global Village makes the more expensive PowerPort/Mercury Duo.)

 Part I
 1
 What Is the Internet?

Suffice it to say that you want the fastest standard modem you can lay your hands on, and as of this writing, the fastest standard means that you want a modem with the magic word *v.34* stamped prominently on its box. That word, which says that the modem supports a certain standard method of transmitting information, ensures that your modem talks to most other modems at a high rate of speed, generally 28,000 bits per second (bps). (This speed, although fast for a modem, doesn't even approach that of a local area network [the standard LocalTalk networking built into every Macintosh runs at 230,000 bps]).

Note

Although v.32bis, the protocol used for 14,000 bps modems, is probably still more common, I recommend that you buy a 28,000 bps modem, given the choice.

Modem manufacturers often make claims about maximum throughput being 57,600 bps or higher, but real speeds vary based on several variables such as phone line quality, compression, and the load on the host. Except in laboratory situations and near black holes, modems never reach the promised maximum speed. The main point to keep in mind is that it takes two to tango; the modems on either end of a connection drop to the slowest common speed (usually 2,400 bps) if they don't speak the same protocols. Just think of this situation as my trying to dance with Ginger Rogers—there's no way she and I could move as quickly as she and Fred Astaire did.

Note

You may see such terms as v.fast and v.terbo (or v.turbo), but remember what I said about needing two to tango. If your provider doesn't use modems that also support these non-standards, there's no point in worrying about anything other than v.32bis or v.34 for right now.

Actually, there are more caveats to the modem question than I'd like to admit. Modems work by converting digital bits into analog waves that can travel over normal phone lines, and—on the other end—translating those waves back into bits. Translation of anything is an inherently error-prone process, as you know if you've ever managed to make a fool of yourself by trying to speak a foreign language.

A large percentage of the problems that I've seen people have since the first edition of this book were related to their modems. Modem troubles are exacerbated by the fact that modem manuals are, without a doubt, the worst excuse for technical writing that I've ever seen. They're confusing, poorly written, poorly organized, and usually concentrate on the commands that the modem understands without providing any information as to what might go wrong. So, as much as I'd like to pretend that modems are all compatible, and that setting one up to communicate with an Internet host is a simple process, it may not work right away. If you encounter problems after first checking all the settings to make sure you've done everything right, you should then check your settings against those in your modem manual. It's also worth asking your Internet provider for suggestions on settings for your modem.

I can't tell you how unhappy I am to have written that last paragraph, but it's just how the world is. You probably didn't get a driver's license without passing a written test, practicing with an adult, taking Driver's Ed, and finally passing a practical test. Perhaps more apt, you probably weren't able to find anything in a school library until one of the teachers or librarians showed you around. If your modem works on the first try, great! If not, don't get depressed—not everything in this life is as easy as it should be. If it were, we'd have world peace.

Anyway, modems connect to phone lines, of course, and residential phone lines are generally self-explanatory, although at some point you may want to get a second line for your modem. Otherwise, those long sessions reading news or downloading the latest and greatest shareware can irritate loved ones who want to speak with you. (Of course, those sessions also keep telemarketers and loquacious acquaintances off your phone.) I also thoroughly enjoy being able to search the Internet for a file, download it, and send it to a friend who needs it, all while talking to him on the phone.

Not all telephone lines are created equal, and you may find that yours suffers from *line noise*, which is static on the line caused by any number of things. Modems employ error correction schemes to help work around line noise, but if it's especially bad, you may notice your modem slowing down as it attempts to compensate for all the static. When

it's really bad, or when someone induces line noise by picking up an extension phone, your modem may just throw up its little modem hands and hang up on you. You can complain to the phone company about line noise; as I understand it, telephone lines must conform to a certain level of quality for voice transmissions. Unfortunately, that level may not be quite good enough for modems, especially in outlying rural areas, but if you're persnickety enough, you can usually get the phone company to clean up the lines sufficiently.

Note

If you connect from home and order a second line from your phone company, don't be too forthcoming about why you want the second line. Although they provide no additional quality or service, business rates are higher than residential rates. Some phone companies are sticky about using modems for non-business purposes, which is why this point is worth mentioning. If you connect to the Internet from your office, there's no way around this situation.

Part I 4 What Is the Internet?

Software

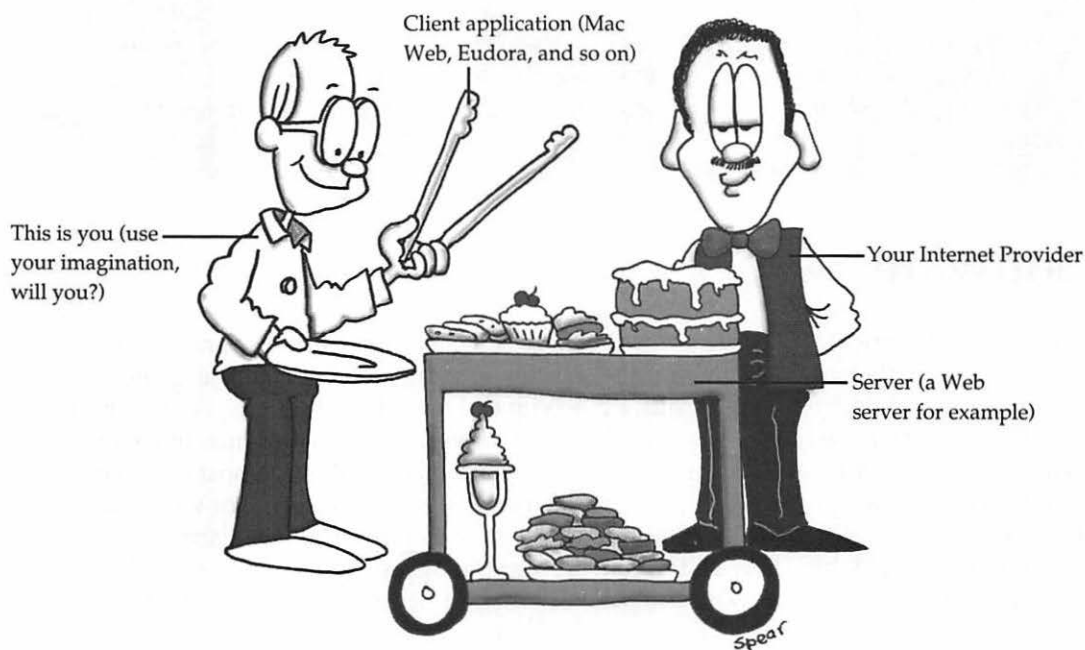
As for the software, the programs that probably come to your mind first are the freeware and shareware files stored on the Internet for downloading—things such as games, utilities, and full-fledged applications. I'll let you discover those files for yourself though, and I'll concentrate on the software available for connecting to the Internet, much of which is free. Other programs are shareware or commercial, although most don't cost much. I'll talk about pretty much every piece of software I know about for working on the Internet in Part IV of this book. Although there's no way for the book's discussion to keep up with the rate at which new and updated programs appear, I provide the latest versions of all the freeware and shareware programs on my file site, **ftp.tidbits.com**. Don't worry about the details now—I'll get to them later in the book.

For the time being though, I want to hammer home a few key points to help you understand, on a more gut level, how this setup all works. First, the Internet machines run software programs all the time. When you use electronic mail or Telnet or most anything else, you are actually using a software program, even if it doesn't seem like it. That point is important because as much as you don't need to know the details, I don't want to mystify the situation unnecessarily. The Internet, despite appearances, is not magic.

Second, because it takes two to tango on the Internet (speaking in terms of host and client machines), a software program is always running on both sides of the connection. Remember the client and host distinctions for machines? That's actually more true of the software, where you generally change the term *host* to the term *server*, which gives the broader term *client/server computing*. So, when you run a program on the Mac, say something like Fetch (an FTP client that retrieves files), it must talk to the FTP server program that is running continually on the remote machine. The same is true no matter what sort

of connection you have. If you're using a Unix command-line account and you run a program called Lynx to browse the World Wide Web, Lynx is a client program that communicates with one or more World Wide Web servers on other machines.

Think of a fancy restaurant where they bring around a dessert cart filled with luscious pastries at the end of your meal. You're not allowed to get your grubby hands on the food itself, so the restaurant provides a pair of dessert tongs that you must use to retrieve your choice of desserts. That's exactly how client/server computing works. The dessert cart is the server—it makes the information, the desserts in this example, available to you, but only via the client program—the dessert tongs. Hungry yet?



Third, FTP and Fetch are the high-level programs that you interact with, but low-level software also handles the communications between Fetch and an FTP server. This communication at multiple levels is how the Internet makes functions understandable to humans and still efficient for the machines, two goals that seldom otherwise overlap.

So, if you can cram the idea into your head that software makes the Internet work on both a high level that you see and a low level that you don't, you'll be much better off. Some people never manage to understand that level of abstraction, and as a result, they never understand anything beyond how to type the magic incantations they have memorized. Seeing the world as a series of magic incantations is a problem because people who do that are unable to modify their behavior when anything changes, and on the Internet, things change every day.

Information

More so than any other human endeavor, the Internet is an incredible, happy accident. Unlike the library at Alexandria (the one that burned down) or the Library of Congress, the Internet's information resources follow no master plan (although the Library of Congress, as do many other large university and public libraries, has its catalog and some of its contents on the Internet). No one works as the Internet librarian, and any free information resources that appear can just as easily disappear if the machine or the staff goes away. And yet, resources stick around; they refuse to die—in part because when the original provider or machine steps down, someone else generally feels that the resource is important enough to step in and take over.

Note

Interestingly, much of the Library of Congress Gopher site was built on volunteer time by government employees. They created and still maintain the site in addition to their regular duties on their own time, because they believe in the principle of the widest possible dissemination of government (and other non-copyrighted) information to the taxpayer.

Andy Williams at Dartmouth, for instance, runs a mailing list devoted to talking about scripting on the Macintosh, specifically about AppleScript and Frontier (an Apple event-based scripting program from UserLand Software that is now starting to become an especially interesting tool in controlling some Internet programs like Eudora and Netscape). Andy also originally made sample scripts and other files pertaining to Frontier available, but he was not able to keep up with the files and still do his real job (a common problem). Luckily, Fred Terry at the University of Kansas quickly stepped in and offered to provide a Frontier file site because he was already storing files related to two other Macintosh programs, QuicKeys and Nisus. (Fred also rescued the Nisus mailing list when Brad Hedstrom, the list creator and administrator, had to bow out, and Fred's probably a sucker for stray dogs, too.) Fred felt that keeping the information available on the Internet was important and that the sacrifice was sufficiently small.

Just to fill in more of the story, Fred was actually running a list about AppleScript on his site, along with lists about QuicKeys and Nisus. When it became clear that discussions about scripting on the Macintosh overlapped both the AppleScript and the Frontier list, Fred and Andy got together and created a single list at Dartmouth—today's MacScripting list—and they've continued to collaborate, with Dartmouth picking up both the Nisus and QuicKeys lists as well.

Andy's something of a sucker for resources in need of a home, as well. When a man named Bill Murphy came up with a method of translating our issues of *TidBITS* into a form suitable for display on the World Wide Web, he ran into the problem of not having a sufficiently capable machine to provide that information to the Internet community. Who should step in but Andy, who offered the use of a World Wide Web server that he runs at

Dartmouth. Between Bill's and Andy's selfless volunteer efforts, the Internet had yet another information resource for anyone to use.

These are just a few examples of the way information can appear on the Internet. Damming the Internet's flow of information would be harder than damming the Amazon with toothpicks. In fact, some of the Internet's resiliency is due to the way the networks themselves were constructed, but we'll get into that later. Next, let's look at the main ways information is provided on the Internet.

The Internet Post Office

You can think of an Internet host machine as a post office, a large post office in a large metropolitan area. In that post office, huge quantities of information are dispensed every day, but it doesn't just gush out the front door. No, you have to go inside, sometimes wait in line, and then go to the appropriate window to talk to the proper clerk to get the information that you want. You don't necessarily pick up mail that's been held for you at the same window as you purchase a money order. Internet information works in much the same way. But on an Internet host, instead of windows, information flows through virtual ports (they're more like two-way television channels than physical SCSI ports or serial ports). A port number is, as I said, like a window in the post office—you must go to the right window to buy a money order, and similarly, you must connect to the right port number to run an Internet application. Luckily, almost all of this happens behind the scenes, so you seldom have to think about it. See Table 3.2 for a list of some common port numbers.

Table 3.2
A Few Common Port Numbers

<i>Port Number</i>	<i>Description</i>
20, 21	File Transfer Protocol (data on 20, control on 21)
23	Telnet
25	Simple Mail Transfer Protocol
53	Domain Name Server
70	Gopher
79	Finger
80	World Wide Web
110	Post Office Protocol - Version 3
119	Network News Transfer Protocol
123	Network Time Protocol
194	Internet Relay Chat Protocol

Note

*I found this information in RFC (Request for Comment) 1340 via the Web or Gopher at **is.internic.net**. All of the RFCs that define Internet standards are stored there should you want more technical information about how parts of the Internet work at their lowest levels.*

 Part I
 4
 What Is the Internet?

So, in our hypothetical Internet post office, there are seven main windows that people use on a regular basis. There are of course hundreds of other windows, usually used by administrative programs or other things that people don't touch much, but we won't worry about them. The main parts to worry about are email, Usenet news, Telnet, FTP, WAIS, Gopher, and the World Wide Web. Each provides access to different sorts of information, and most people use one or more to obtain the information they need.

Now that I've said how they're all similar, in the sense of all working through connections to the proper ports, there are some distinctions we must make between the various Internet services.

Email and Usenet news (along with MUDs and Internet Relay Chat) are forms of interpersonal communication—there is always a sender and a recipient. Depending on the type of email message or news posting, you can use different analogies relating to the paper world, and I'll get to those in a moment.

All of the information made available through other main parts of the Internet, such as Telnet, FTP, WAIS, Gopher, and the World Wide Web, is more like information in libraries than interpersonal communication, in the sense that you must visit the library specifically, and once there, browse or search through the resources to find a specific piece of information. These services have much more in common with traditional publishing than email and news.

Note

I should note that, in my eyes, the difference between browsing and searching is merely that when you're browsing, you're not looking for a specific piece of information. Perhaps you only want some background, or simply want to see what's out there. When you're searching, you usually have a particular question that you want answered.

No matter what you use, there is still some sort of communication of information going on. With email and news, it's generally informal and between individuals, whereas with the rest of the Internet services, the information is usually more distilled—that is, someone has selected and presented it in a specific format and in a specific context. None of these distinctions are hard and fast. Much informal information is available via Gopher, for instance, and it's certainly easy enough to find distilled information via email. I'll try to give you a sense of what each service is good for when talking about them later on.

Electronic Mail

Email is used by the largest number of people on the Internet, although in terms of traffic, the heaviest volumes lie elsewhere. Almost everyone who considers herself connected to the Internet in some way can send and receive email.

As I said previously, most personal exchanges happen in email because email is inherently an interpersonal form of communication. All of your email comes into your electronic mailbox, and unless you let them, no one else can easily read your mail. When you get a message from a friend via email, it's not particularly different than getting that same message, printed out and stuffed in an envelope, via snail mail. Sure, it's faster and may have been easier to send, but in essence personal email is just like personal snail mail.

Because it's trivial to send the same piece of email to multiple people at once, you also can use email much as you would use snail mail in conjunction with a photocopy machine. If you write up a little personal newsletter about what's happening in your life and send it to all your relatives at Christmas, that's the same concept as writing a single email message and addressing it to multiple people. It's still personal mail, but just a bit closer to a form letter.

The third type of email is carried on *mailing lists*. Sending a submission to a mailing list is much like writing for a user group or alumni newsletter. You may not know all of the people who will read your message, but it is a finite (and usually relatively small) group of people who share your interests. Mailing list messages aren't usually aimed at a specific person on the list, but they are more intended to discuss a topic of interest to most of the people who have joined that list. However, I don't want to imply that posting to a mailing list is like writing an article for publication because the content of most mailing lists more resembles the editorial page of a newsletter than anything else. You'll see opinions, rebuttals, diatribes, questions, comments, and even a few answers. Everyone on the list sees every posting that comes through, and the discussions often become quite spirited.

The fourth type of email most resembles those "bingo cards" that you find in the back of many magazines. Punch out the proper holes or fill in the appropriate numbered circle, return the card to the magazine, and several weeks later you'll receive the advertising information that you requested. For instance, I've set up my Macintosh to send an informational file about *TidBITS* automatically to anyone in the world who sends email to a certain address (**info@tidbits.com**, if you're impatient and want to try something right away). A number of similar systems exist on the Internet, dispensing information on a variety of subjects to anyone who can send them email. A variant of these auto-reply systems is the *mailserver*, which generally looks at the Subject line in the letter or at the body of the letter and returns the requested file. Mailservers enable people with email-only access to retrieve files that otherwise are available only via FTP.

Usenet News

Like email-based discussion lists, Usenet news is interpersonal information—it comes from individuals and is aimed at thousands of people around the world. Unlike email, even unlike mailing lists, you cannot find out who makes up your audience. Because of this unknown audience, posting a message to Usenet is more like writing a letter to the editor of a magazine or major metropolitan newspaper with hundreds of thousands of readers. We have ways of estimating how many people reach each of the thousands of Usenet groups, but the estimates are nothing more than statistical constructs (though hopefully accurate ones).

Almost everything on Usenet is a discussion of some sort, although a few groups are devoted to regular information postings, with no discussion allowed. The primary difference between Usenet news and mailing lists is that news is more efficient because each machine receives only one copy of every message. If two users on the same machine (generally multi-user mainframes or workstations at this point) read the same discussion list via email, getting the same information in news is twice as efficient. If you have a large mainframe with 100 people all reading the same group, news suddenly becomes 100 times as efficient because the machine stores only the single copy of each message, rather than each individual receiving her own copy.

In many ways, Usenet is the kitchen table of the Internet—the common ground where no subject is taboo and you must discuss everything before implementing it. In great part because of the speed at which Usenet moves (messages appear quickly and constantly, and most machines don't keep old messages for more than a week due to lack of disk space), finding information there can be difficult. Think of Usenet as a river, and you must dip in to see what's available at a specific point in time because that information may disappear downstream within a few days.

Note

The speed at which messages disappear from Usenet varies by group and by the machine you use. Each administrator sets how long messages in a group will last before being expired or deleted from the system. Messages in newsgroups with many postings per day may expire after a day or two; messages in groups with only a few postings per week may last a month. Because complete Usenet traffic is about 200 MB of information each day, you can see why the short expiration times are essential.

You can, of course, always ask your own question, and you usually get an answer (though it may be one you don't like), even if it's the sort of question everyone asks. Common questions are called Frequently Asked Questions, or FAQs, and are collected into lists and posted regularly for newcomers. Luckily, the cost of disk storage is decreasing sufficiently so that some people and organizations are starting to archive Usenet discussions. These enable you to use search engines like WAIS to go back and search for information that flowed past in a mailing list or newsgroup a long time ago.

Telnet

Telnet is tough to describe. The best analogy I can think of is that Telnet is like an Internet modem. As with a standard modem, Telnet enables your computer to communicate with another computer somewhere else. Where you give your modem a phone number to dial, you give Telnet an Internet address to connect to. And just like a modem, you don't really do anything within Telnet itself other than make the connection—in the vernacular, you telnet to that remote computer. Once that connection is made, you're using the remote computer over the Internet just as though it were sitting next to you. This process is cool because it enables me to telnet to the mainframes at Cornell University, for example, and use them just as I did when I was actually in Ithaca, and not 3,000 miles away in Seattle.

Note

Telnet, FTP, and Gopher can all work as both nouns describing the service or the protocols, and as verbs describing the actions you perform with them. If Telnet or Gopher is capitalized in this book, it's a noun describing the service; if it's in lowercase, it's a verb describing the action. (FTP is always capitalized in the book, because it's an acronym.) Unfortunately, others on the Internet aren't as consistent (and they don't have editors checking their text), so this isn't a universal convention.

I realize that I'm supposed to talk about information in this section, but Telnet is such a low-level protocol that it's impossible to separate the information that's available via Telnet from the protocol itself.

Most people don't have personal accounts on machines around the world (and I never use the Cornell mainframes anymore either), but a number of organizations have written special programs providing useful information that anyone can run over the Internet via Telnet.

Say I want to search for a book that's not in my local library system. I can connect via Telnet to a machine that automatically runs the card catalog program for me. I can then search for the book I want, find out which university library has it, and then go back to my local library and ask for an inter-library loan.

Or, for a more generically useful example, if you telnet to **downwind.spr1.umich.edu 3000**, you reach the University of Michigan's Weather Underground server, with gobs of data about the weather around the entire country.

File Transfer Protocol (FTP)

FTP feels like it's related to Telnet, but in fact that's an illusion—the two are basic protocols on the Internet, but are otherwise unrelated. While Telnet simply enables you to connect to another remote computer and run a program there, FTP enables you to connect to a remote computer and copy files back and forth. It's really that simple.

More data is transferred via plain old FTP than by any other method on the Internet, and it's not surprising because it's a least-common denominator that almost every machine on the Internet supports. Like Telnet, you must be directly connected to the Internet while using FTP, although there are a few special FTP-by-mail services that enable you to retrieve files stored on FTP sites by sending specially formatted email messages to an FTP-by-mail server.

There are probably millions of files available via FTP on the Internet, although you may discover that many of them are duplicates because people tend to want to give users more than one way to retrieve a file. If a major file site goes down for a few days, it's nice to have a *mirror site* that has exactly the same files and can take up the slack.

Note

Mirror sites are important because as the Internet grows, individual machines become overloaded and refuse to accept new connections. As with anything that's busy (like the phone lines on Mother's Day, the checkout lines at 5:00 P.M. on Friday afternoons at the grocery store, and so on) it always seems that you're the one who gets bumped or who has to try over and over again to get through. Don't feel special—hundreds of other people suffer exactly the same fate all the time. Mirror sites help spread the load.

In the Macintosh world, several sites with lots of disk space (several gigabytes, actually) store a tremendous number of freeware and shareware programs along with commercial demos and other types of Macintosh information. If you think your local BBS has many files, wait until you see the two main file sites for the Mac, **mac.archive.umich.edu** and the Info-Mac Archive site currently at **sumex-aim.stanford.edu**. I would estimate that between 20 and 100 new files appear in the Info-Mac Archive archive every week—certainly enough to keep you busy if you enjoy exploring freeware and shareware software for the Macintosh.

The vastness of the number of files stored on FTP sites may stun you, but you have access to a tool that helps bring FTP under control. Archie takes the grunt work out of searching numerous FTP sites for a specific file. You ask Archie to find files with a specific keyword in their names, and Archie searches its database of many FTP sites for matches. Archie then returns a listing to you, providing the full file names and all the address information you need to retrieve the file via FTP.

Note

If you have MacTCP access to the Internet, you can use a program I included on the disk called Anarchie, which is an FTP and Archie client program for the Mac. Just ask Archie to find a file, and when the results come back, double-click on the file to retrieve it. And people complain about how hard the Internet is!

WAIS

I mentioned using WAIS to search for information about deforestation in the Amazon rainforest in the preceding chapter, but that's only the tip of the iceberg. WAIS originated from a company called Thinking Machines, but has now split off into its own company, WAIS, Inc. Using the tremendous processing power of Thinking Machines' Connection Machine supercomputer or another powerful computer, WAIS can quickly (usually under a minute) return a number of articles to English-language queries, sorted by the likelihood that they are relevant to your question. WAIS is limited only by the information that people feed into it.

Last I counted, there were over 700 sources available for searching within topics as diverse as Buddhism, cookbooks, song lyrics, Supreme Court decisions, science fiction book reviews, and President Clinton's speeches. For all the sources on non-technical topics, I'm sure an equal number exist about technical topics in many fields.

Note

People talking about WAIS (pronounced "ways," I hear) tend to use the terms "source," "server," and "database" interchangeably, and so do I.

Perhaps the hardest part about WAIS is learning how to ask it questions. Even though you can use natural English queries, it takes your question quite literally, and only applies it to the selected sources. So, if you asked about deforestation in the Amazon rainforest while searching in the Buddhism source, I'd be surprised if you found anything.

Since the rise of the World Wide Web's popularity, WAIS seems to have faded into the background a bit. It's not that WAIS servers aren't still being used heavily, but they're being used as search engines behind the scenes, usually through a Web page interface.

Gopher

Gopher, which originated with the Golden Gophers of the University of Minnesota, is an information browser along the same lines as FTP, but with significant enhancements for ease of use and flexibility. Numerous sites—over 2,300 at last count—on the Internet run the host Gopher software, placing information in what are colloquially called *gopher holes*.

When you connect to a Gopher site, you can search databases, read text files, transfer files, and generally navigate around the collection of gopher holes, which is itself called *Gopherspace*.

Gopher can be quite useful in terms of actually making information available that you need to answer specific questions. Part of the reason for this opinion is *Veronica*, and to a lesser extent *Jughead*, which enable you to search through Gopherspace as Archie enables you to search for files on anonymous FTP servers.

Note

Veronica and Jughead were both named to match Archie (from the Archie comics), but Veronica's creators at the University of Nevada did come up with an acronym as well—Very Easy Rodent-Oriented Net-wide Index to Computerized Archives. Jughead stands for Jonzy's Universal Gopher Hierarchy Excavation And Display. Glad you asked?

Veronica searches through all of Gopherspace, which is useful, although badly phrased searches (Veronica doesn't use natural English, as WAIS does) can result in hundreds of results. Jughead searches a subset of Gopherspace and can thus be more accurate, though less comprehensive.

One of the special features of Gopher is that it provides access to FTP (and Archie) and WAIS, and can even run a Telnet program to provide access to resources only available via Telnet. Gopher also can work with other programs to provide access to special data types, such as pictures and sounds. When you double-click on a picture listing in Gopher, it downloads the file and then runs another program to display the picture. This sort of integration doesn't generally work all that well if all you have is Unix command-line access to the Internet.

World Wide Web

When I wrote the first edition of this book in mid-1993, the World Wide Web existed, but lacked a good client program on the Macintosh. I managed to write a paragraph or two back then about NCSA Mosaic, the Web browser that was officially released a few months after I finished the book, but there simply wasn't much on the Web at that point. Now there are at least four well-known Web browsers for the Macintosh alone, and I'm sure more are on the way. Everything about the Web has changed since that first edition. It's become much, much larger, and the resources available on it have become incredibly diverse and far more useful.

Note

You may see the World Wide Web referred to as simply "the Web," "W3," or sometimes as "WWW."

The Web brings a couple of very important features to the Internet. First, unlike Gopher or anything else, it provides access to full fonts, sizes, and styles for text, and can include images onscreen with no special treatment. Sounds and movies are also possible, though often too large for many people to download and view. Second, the Web provides true hypertextual links between documents anywhere on the Web, not just on a single machine. For those unfamiliar with *hypertext*, it's a powerful concept that enables you to navigate flexibly through linked pieces of information. If you read a paragraph with a link promising more information about the topic, say results from the last Olympic Games, simply click on the link, and you'll see the results. It really is that simple, and the World Wide Web enjoys the highest profile of any of the Internet services.

There are more machines whose names start with **www** than anything else now, and the Web is in second place and rapidly catching up with FTP in terms of the amount of data transferred. Nothing touches it in terms of pure sexiness, although many Web servers that you see suffer from the same problem that many publications did after the Macintosh made desktop publishing popular: they're designed by amateurs and are ugly as sin.

Summing Up

I've tried to answer one of the harder questions around, "What is the Internet?" The simple answer is that the Internet is a massive collection of people, machines, software programs, and data, spread all around the world and constantly interacting. That definition, and the explication I've provided about the various parts of the Internet elephant, should serve you well as we look next at the history of this fascinating beast.

Chapter 5



The Internet Beanstalk

Unlike the Greek goddess Athena, the Internet did not spring from the head of some Zeusian computer scientist. It was formed by a process of relatively rapid accretion and fusion (but keep in mind that this industry is one in which computer power doubles every few years). In 1980, there were 200 machines on the Internet—that number is now about 4.9 million. The grain of sand that formed the heart of this giant electronic pearl came from the U.S. Department of Defense (DoD) in 1969. I'm pleased to be older than the Internet, having been born in 1967, but I'm not enough older to talk authoritatively about world conditions at that time. So, please bear with my second-hand retelling.

Cold War Network

In the 1950s, the Russian Sputnik program humiliated the United States. To better compete in the space race, the U.S. space program (at the time under the auspices of the military) received major government funding. That funding came from the DoD under its Advanced Research Projects Agency (ARPA). In the early 1960s, the space program left the military to become NASA, but ARPA remained, and as with many government programs that have seemingly lost their reason to exist, so did its funding. What to do with the money?

The DoD was, at that time, the world's largest user of computers, so J.C.R. Licklider and others proposed that ARPA support large-scale basic research in computer science. ARPA didn't originally require that the research it supported be either classified or directly related to military applications, which left the door open for far-reaching research in many fields. In 1963, ARPA devoted a measly \$5 to \$8 million to its computer research, the Information Processing Technologies Office (IPTO), first under Licklider, and then subsequently under the 26-year-old Ivan Sutherland, who had developed an early (perhaps the earliest) graphics program at MIT. After Sutherland, a 32-year-old named Robert Taylor headed IPTO. Taylor managed to double IPTO's budget at a time when ARPA's overall budget was decreasing, and even admitted to diverting funds from military-specific projects to pure computer science.

Around this time, the ARPAnet (Advanced Research Projects Agency Network) began connecting various computers around the country at sites performing research for ARPA. Computers were expensive, and sharing them was the only way to distribute the resources appropriately. Distribution of cost via networks proved to be an important force in the development of the Internet later on as well. Proponents like Taylor ensured the early survival of the fledgling ARPAnet when it was all too vulnerable to governmental whimsy.

In 1969, Congress got wind of what ARPA was up to in terms of funding basic research with money from the defense budget. Three senators, including the still-active Edward Kennedy, pushed through legislation requiring that ARPA show that its programs were directly applicable to the military. In the process, ARPA's name changed to reflect its new nature; it became the Defense Advanced Research Projects Agency, or DARPA. (Years later, the name changed back to ARPA again, just to confuse the issue.) Bob Taylor became entangled in some unpleasant business reworking military computers in Saigon during the Vietnam War and left DARPA shortly thereafter. He was succeeded by Larry Roberts, who played a major role in getting the then two-year-old ARPAnet up and running. Stewart Brand, founder of *The Whole Earth Catalog*, wrote at the time:

*At present some 20 major computer centers are linked on the two-year-old ARPAnet. Traffic on the Net has been very slow, due to delays and difficulties of translation between different computers and divergent projects. Use has recently begun to increase as researchers travel from center to center and want to keep in touch with home base, and as more tantalizing sharable resources come available. How Net usage will evolve is uncertain. There's a curious mix of theoretical fascination and operational resistance around the scheme. The resistance may have something to do with reluctance about equipping a future Big Brother and his Central Computer. The fascination resides in the thorough rightness of computers as communication instruments, which implies some revolutions. (Stewart Brand, in *II Cybernetic Frontiers*, Random House, 1974)*

So if DARPA had to justify the military applications of its research, what survived? Well, the ARPAnet did, and here's why: As leaders of the free world (pardon the rhetoric), we needed the latest and greatest methods of killing as many other people as possible. Along

with *offensive* research must perforce come *defensive* research; even the DoD isn't so foolish as to assume we could wage a major war entirely on foreign soil. For this reason, the tremendous U.S. interstate highway system served double duty as a distribution medium for tanks and other military hardware. Similarly, the Internet's precursor was both a utilitarian and experimental network. ARPAnet connected both military research sites (hardware was expensive and had to be shared) and was an experiment in resilient networks that could withstand a catastrophe—including, in the imaginations of the DoD planners of the day, an atomic bomb.

Interestingly, the resiliency of the ARPAnet design, as carried down to the Internet, has led some to note that the Internet routes around censorship as it would route around physical damage. It's a fascinating thought, especially in regard to Stewart Brand's earlier comment about Big Brother. If anything, the Internet actually has served to reduce the threat of a Big Brother, because it makes communication between people so fluid and unrestricted. But, I anticipate myself.

Gateways

As a result of the machinations described previously, the Internet Protocol, or IP (the second half of TCP/IP) was created. Essentially, the point behind IP systems is that each computer knows of, or can determine, the existence of all the others, and thus route packets of information to its destination via the quickest route. While doing this, they are able to take into account any section of the network that's been bombed out or has merely been cut by an over-enthusiastic telephone repairperson. This design turns out to work well; more importantly, it makes for an extremely flexible network. If your computer can get a properly addressed packet of information to a machine on the Internet, that machine will worry about how to deliver it, translating as necessary. That's the essence of a *gateway*—it connects two dissimilar networks, translating information so that it can pass transparently from one to the other.

In the early 1980s, the military began to rely more and more heavily on the ARPAnet for communication, but because the ARPAnet still connected a haphazard mix of research institutions, businesses doing defense work, and military sites, the military wanted their own network. And so the ARPAnet split in half, becoming the ARPAnet and the Milnet (Military Network). The ARPAnet continued to carry traffic for research sites, and even though the military now had its own Milnet, traffic passed between the ARPAnet and the Milnet by going through gateways.

The concept of gateways proved important in the history of the Internet. Alongside the development of the Internet came the development of a number of other, generally smaller, networks that used protocols other than IP, such as BITNET, JANET, and various others. These also included some like Usenet and CSNET that didn't care what protocols were used. These networks were regional or dedicated to serving certain types of machines or users.

Perhaps the largest driving force behind the Internet is that of the need to connect with other people and other networks. The grass is always greener on the other side of the fence, and gradually gateway sites sprung up so that email could pass among the different networks with ease.

Usenet

I'm going to take a brief break from the Internet itself, because at approximately the same time the ARPAnet split, a host of other networks came into being, probably the most interesting of which was Usenet, the User's Network.

Usenet started in 1979, when some graduate students decided to link several Unix computers together in an attempt to better communicate with the rest of the Unix community. The system they created included software to read news, post news, and transport news among machines. To this day, that simple model continues, but whereas once two machines were on Usenet, today there are hundreds of thousands. The software that transports and displays Usenet news now runs on not just Unix machines, but on almost every type of computer in use on the networks. The topics of discussion have blossomed from Unix into almost any conceivable subject—and many inconceivable ones. Like all the other network entities, Usenet quickly grew to be international in scope and size.

Unlike many of the other networks, Usenet truly grew from the bottom up, rather than from the top down. Usenet was created by and for users, and no organization—commercial, federal, or otherwise—had a hand in it originally. In many ways, Usenet has provided much of the attitude of sharing that exists on the Internet today. In the past, you usually got a Usenet *feed* (that is, had another machine send news traffic to your machine) free of charge (other than your telephone charges) as long as you were willing to pass the feed on to someone else free of charge. Due to commercial pressures, the days of the free feeds are essentially no more, but the spirit of cooperation they engendered remains in much of what happens on the Internet.

I don't want to imply that Usenet is this happy, carefree network where everything is free and easy, because in many cases it's a noisy, unpleasant network that exists because of the utility of some of the information that it carries. Despite the attitude toward sharing, the survival of Usenet is due in large part to the resourcefulness of network administrators at major sites. Faced with mounting telephone charges for long distance calls between Usenet hosts, these administrators found a way to carry Usenet news over the TCP/IP-based Internet rather than just the previous modem-based UUCP connections. Thus, they prevented the costs of carrying Usenet from coming to the attention of the bean counters poised to strike unnecessary expenses from their budgets. The TCP/IP connections of the ARPAnet, and then the Internet, were already paid for. So, by figuring out how to carry Usenet over those lines, the network administrators managed to cut their costs, keep users happy, and save Usenet from itself in the process. In other words, Usenet may be an anarchy, but it wouldn't stand a chance without some occasional help from high places.

BITNET

Shortly after Usenet took its first faltering networked steps, Ira Fuchs of City University of New York and Greydon Freeman of Yale University decided to network their universities using IBM's then-new NJE communications protocol. Although this protocol later expanded to support Digital Equipment's Vaxen running VMS and even some implementations of Unix, the vast majority of machines on BITNET (the "Because It's Time" network) have always been IBM mainframes. Fuchs and Freeman made their connection in the spring of 1981. BITNET grew rapidly, encompassing over 100 organizations on 225 machines by 1984, and by 1994 reaching the level of 1,400 organizations in 49 countries around the world. Most BITNET sites are at universities, colleges, and other research institutions.

BITNET has always been a cooperative network; members pass traffic bound for other sites free of charge, and software developed by one has been made available to all. Unlike Usenet, however, BITNET developed an organizational structure in 1984. This took the form of an Executive Committee, made up of representatives of all the major nodes on the network. Also in 1984, IBM presented a large grant that provided initial funding for centralized network support services. This grant, coupled with the fact that most of the machines on BITNET were IBM mainframes, gave rise to the erroneous rumor that BITNET was an IBM network. In 1987, BITNET became a nonprofit corporation. In 1989, it changed its corporate name to CREN, the Corporation for Research and Educational Networking, when it merged its administrative organization with another of the parallel educational networks, CSNET (the Computer+Science Network). Today, BITNET is in something of a decline, due in large part to the nonstandard NJE protocol in an increasingly IP world.



NSFNET

The next big event in the history of the Internet was the creation of the high-speed NSFNET (National Science Foundation Network) in 1986. NSFNET was developed to connect supercomputer sites around the country. Because supercomputers are terribly expensive, the NSF could afford to fund only five (and even then they received some major financial help from companies like IBM). With this limited number, it made sense to network the supercomputers so that researchers everywhere could use them without traveling great distances. At first, the NSF tried to use the ARPAnet, but that attempt quickly became bogged down in bureaucracy and red tape.

The NSF therefore decided to build its own network. Merely connecting the five supercomputer sites wasn't going to help the vast majority of researchers, of course, so the NSF created (or used existing) regional networks that connected schools and research sites in the same area. Then those networks were connected to the NSFNET.

To quote from W.P. Kinsella's *Shoeless Joe*, "If you build it, they will come." Perhaps not surprisingly, once all of these networks were able to communicate with one another, the supercomputer usage faded into the background. Other uses, most notably email, became preeminent. One of the important features of the NSFNET was that the NSF encouraged universities to provide wide access to students and staff, so the population of and traffic on the net increased dramatically.

In 1987, the NSF awarded a contract to a group of companies to manage and upgrade the NSFNET. This group was made up of IBM, MCI, and Merit Network, which ran the educational network in Michigan. The group dealt with the massive increase in traffic by replacing the old lines with much faster connections.

Eventually the NSFNET had entirely supplanted the ARPAnet, and in March of 1990, the ARPAnet was taken down for good, having played the starring role for 21 years. Similarly, another national network, CSNET, which had connected computer science researchers around the country, closed its electronic doors a year later, all of its traffic having moved to the faster NSFNET.

NREN

The NSFNET is all fine and nice, but in many ways it discriminated against "lower" education—two-year colleges, community colleges, and the much-maligned K-12 schools. To save the day, then-Senator Al Gore sponsored a bill, passed in December of 1991, called the "High-Performance Computing Act of 1991." Gore's legislation created a new network on top of (and initially using) the NSFNET. This new network is called the NREN, for National Research and Education Network. Along with providing even faster speeds, the NREN specifically targets grade schools, high schools, public libraries, and two- and four-year colleges. In working with the thousands of people who subscribe to *TidBITS*, I see a lot of email addresses, and it's clear to me that these educational institutions are joining the Internet in droves. A day rarely passes when I don't see something from someone whose address clearly labels him or her as a teacher at a grade school or even a student in a high school.

Alert readers probably have noticed that NREN looks a lot like CREN, and in fact, the acronyms are similar—with reason. CREN recognizes the need for an integrated National Research and Education Network. In fact, as the IBM-created NJE protocol gradually disappears in favor of the more powerful and popular IP, CREN has said it will disband, merge with NREN, or cooperate with it as appropriate—though only when NREN exists with access rules, funding, and usage policies that allow a clean transition. Currently, CREN feels that the NREN does not provide consistent policies regarding these issues.

Who Pays?

More and more of the Internet is being created and run by commercial organizations. All a commercial provider has to do is to pay for its part of the network, just as universities pay for their connections and government departments pay for theirs. The difference is that unlike universities or government organizations, commercial providers want to make money, or at least break even, so they in turn sell access to their machines or networks to other providers or to end users.

The gut reaction to the commercialization of the Internet from the old-timers (who remember when you could get a Usenet feed merely by asking) is often negative, but most people believe that the Internet must accept commercial traffic. In part, this response is true because the only alternative to accepting commercial traffic is actively rejecting it, and no one wants to sit around censoring the Internet, were that even possible. In many ways, the question has already been decided because there are now more **com** addresses than **edu** addresses, the previous champ.

Commercialization also allows small organizations to create the equivalent of wide-area networks that previously only large businesses could afford. A company such as Microsoft can spend the money to install an international company network, but few companies are so large or so wealthy. Many may not need such an international network, but they may need enhanced communications. Email can be a powerful medium for business communication, just as it is for personal communication. And, if transferring a file via FTP or email can save a few uses of an overnight courier, the connection can pay for itself in no time.

In addition, whereas in the past you had to work at a large business or university to gain Internet access, it has become far easier for an individual to get access without any such affiliation, although the costs are, of course, more obvious. Easier independent access couldn't have happened without increased participation by commercial interests.

The commercialization issue has another side. The U.S. government still controls the NREN, which is a large portion of the Internet and connects many of the major educational sites. As more commercial providers get into the business, the government is gradually relying more and more on them rather than duplicating their effort. This move has much support because the commercial providers can then make money, which is what they want to do, and the government can save money, which is what many people want the government to do.

The concern is, of course, that these commercial providers will want to make too much money and will raise rates significantly or start charging usage-based fees rather than the flat-rate fees that have been more common for this type of service. I don't worry too much about this concern, mostly because the competition is heavy in the Internet field, and



prices are seldom high in extremely competitive fields. In addition, it's not as though the federal funding was necessarily a significant part of the Internet budgets at many institutions.

When these issues started to come to a head in 1994, M. Stuart Lynn, then the head of Cornell Information Technologies, noted that the Internet is a global network, and some countries, such as New Zealand, already have usage-based pricing. So even if the NREN moved to usage-based pricing, most of the global Internet wouldn't be affected. Stuart Lynn also commented that the federal subsidy is trivial to many institutions, and at Cornell it is equivalent to two cans of beer per student per year. In other words, even if Cornell had to rely on a completely commercial network (which may or may not be usage-based), its costs would not change noticeably.

Note

It's worth noting that people like flat-rate fees for most things (telephone service and cable service come to mind), and in the past most personal Internet accounts from commercial providers have been usage-based, with a recent trend toward flat-rate service in the past few years. I believe the increasing number of flat-rate PPP and SLIP accounts from various commercial providers was helped in part by the first edition of this book, with its flat-rate offer for SLIP access from Northwest Nexus. I'm unaware of any other widely available flat-rate accounts that predate the offer from Northwest Nexus. Of course, I could be wrong, but I like to think I had a positive influence.

The trick is to remember that someone always pays for the Internet. If you have a free Internet account thanks to your school, remember that the institution is paying for that connection and funding it in part from your tuition. If your workplace offers Internet access and doesn't limit your use of it, consider that a benefit of working there, along with retirement and health benefits. And an increasingly large number of people, like me, pay directly, usually somewhere between \$5 and \$30 per month. Sure beats cable television.

Remember how I previously said that the NSFNET was created to carry supercomputer traffic but soon found itself being used for all sorts of tasks? That's another basic principle to keep in mind about how the Internet is funded. The network links were created for a specific reason (supercomputer access), and because of that reason, the money necessary to create and maintain those links was allocated in various budgets. Thus, when traffic unrelated to the supercomputer access travels on the same network, it's piggy-backing on the lines that have already been paid for out of existing budgets. So it seems free, and as long as the ancillary traffic doesn't impinge on the supercomputer access, no one is likely to complain. It's much like using your friend's Mac's processing power to generate processor-intensive pictures when he's not using his Mac. As long as your use doesn't slow down the things he wants to do, he probably won't mind, especially if it helps you finish your work sooner. But, if your use prevents him from doing his own work, he'll probably become less generous about it.

So, when the NREN moves completely from governmental to private control, most people won't see the difference because their organizations would continue to foot the bill, especially if the costs didn't change. To poorly funded organizations such as grade schools and public libraries, which may only be able to afford their Internet connections with help from the government, the danger is there. Oh, and where do you think the government gets the money? Taxes, of course. So you end up paying one way or another.

Politics

After all of this discussion, you're probably confused as to who runs what on the Internet. Good, that's the way it should be, because no one person or organization runs the Internet as such. I think of the Internet as a collection of fiefdoms that must cooperate to survive. The fiefdoms are often inclusive as well, so one group may control an entire network, but another group controls a specific machine in that network. As a user, you must abide by what both of them say, or find another host.

I don't mean to say that there aren't some guiding forces. The NSF exercised a certain influence over much of the Internet because it controlled a large part of it in the NSFNET. Thus, the NSF's Acceptable Use Policies (which state that the NSFNET may not be used for "commercial activities") became important rules to follow, or at least keep in mind, and I'll bet that many commercial providers used them as a starting point for creating their own less restrictive, acceptable use policies.

Several other important groups exist, all of which are volunteer-based (as is most everything on the Internet). The Internet Architecture Board, or IAB, sets the standards for the Internet. Without standards, the Internet wouldn't be possible because so many types of hardware and software exist on it. Although you must be invited to be on the IAB, anyone can attend the regular meetings of the Internet Engineering Task Force, or IETF. The IETF's meetings serve as a forum to discuss and address the immediate issues that face the Internet as a whole. Serious problems, or rather problems that interest a sufficient number of volunteers, result in working groups that report back to the IETF with a recommendation for solving the problem. This system seems haphazard, but frankly, it works, which is more than you can say for certain other organizations we could probably name.

Other networks undoubtedly have their controlling boards as well, but the most interesting is Usenet, which has even less organization than the Internet as a whole. Due to its roots in the user community, Usenet is run primarily by the community, as strange as that may sound. Every network administrator controls what news can come into her machine, but she can't control what goes around her machine. The converse applies as well—if a sufficient number of network administrators don't approve of something, say a news-group creation, then it simply doesn't happen. Major events on Usenet must have sufficient support from a sufficient number of people.

Of course, some people's votes count more than others. These people are sometimes called *net heavies* because they often administer major sites or run important mailing lists. The net heavies consider it their job (who knows how they manage to keep real jobs with all the work they do here) to keep the nets running smoothly. Even though they often work behind the scenes, they do an excellent job. Shortly after I started *TidBITS*, for instance, I was searching for the best ways to distribute it. I wasn't able to run a mailing list from my account at Cornell, and *TidBITS* was too big to post to a general Usenet group every week. After I spoke with several of the net heavies, they allowed me to post to a moderated newsgroup, **comp.sys.mac.digest**, that had up to that point been used only for distributing the Info-Mac Digest to Usenet.

If you want to get involved with what organization there is on the Internet, I suggest that you participate and contribute to discussions about the future of the nets. Gradually, you'll learn how the system works and find yourself in a position where you can help the net continue to thrive.

You should keep one thing in mind about the Internet and its loose controlling structure: It works, and it works far better than do most other organizations. By bringing control down to almost the individual level but by requiring cooperation to exist, the Internet works without the strong central government that most countries use and claim is necessary to avoid lawlessness and anarchy. Hmm...

The Internet makes you think, and that's good.

Oh, and remember Bob Taylor, one of the early heads of ARPA's IPTO? Several years later he helped found Xerox PARC, which employed luminaries such as Douglas Engelbart (inventor of the mouse), Alan Kay (a current Apple Fellow), Bob Metcalfe (inventor of Ethernet), and Larry Tesler (who was, and may still be, head of Apple's Advanced Technology Group). And, of course, Xerox PARC was where Steve Jobs saw the Xerox Alto workstation and its graphical interface, and at that moment, in many respects, the Macintosh was born.

The Future

I hope this chapter has provided a coherent view of where the Internet has come from, along with some of the people and networks that were instrumental in its growth. After any history lesson, the immediate question concerns the future. Where can we expect the Internet to go from here?

I'm an optimist. I'm sure you can find someone more than happy to tell you all the horrible problems—technical, political, and social—facing the Internet. I don't hold with such attitudes, though, because something that affects so many people around the world didn't appear so quickly for no reason. In one way or another, I think most people understand on a visceral level that the Internet is good, the Internet is here to stay, and if they want to be someone, they would do well to get access today and contribute in a positive fashion. Of course, books like this one only encourage such utopian attitudes.

In any event, I predict that the Internet will continue growing at an incredible rate. You might make an argument for the rate of growth slowing from its 15 percent per month rate based on the fact that it's silly to assume that anything can continue to grow at such a breakneck speed. A naysayer also might point at the massive influx of novices as endangering the Internet, or point at the increased level of commercialization as a major problem. I feel that such growth is self-propelling and that bringing more people and resources onto the Internet only further fuels the expansion. I think that growth is good—the more people, the more resources, the more opinions, the better off we all are.

I also expect to see the Internet continue to standardize, both officially and informally. At lower levels, more computers will start to use IP instead of BITNET's NJE or the aging UUCP protocols. It's merely a matter of keeping up with the Joneses, and the Joneses are running IP. At a higher level, I think that using various network resources will become easier as they start migrating toward similar interfaces. Just as it's easy to use multiple applications on a Mac because you always know how to open, close, save, and quit, so it will be easier to use new and enhanced services on the Internet because they will resemble each other more and more. Even now, people rely heavily on network conventions such as prefixing site names to indicate what services they provide, like **ftp.tidbits.com** for FTP, **gopher.tidbits.com** for Gopher, and **www.tidbits.com** for the World Wide Web.



And yes, I fully expect to see the Internet become increasingly commercial, both in terms of where the service comes from and in terms of the traffic the Internet carries. However, we must remember the old attitudes about commercial use of the Internet. In the past, commercial use was often acceptable if it wasn't blatant, was appropriately directed, and was of significant value to the readers. In other words, I'll be as angry as the next person if I start receiving automatically generated junk email every day, just as I receive junk mail via snail mail. If such things start happening, the course of action will be the same as it always has been: politely ask the originator to stop once, and then, if that doesn't work, flame away—that is, send back an outrageously nasty message.

Even though I'm optimistic, I know that problems will occur. For example, consider the so-called Green Card debacle. In the spring of 1994, the husband and wife law firm of Canter & Siegel posted a blatantly commercial message advertising a Green Card lottery and immigration services. That wasn't the problem. The problem was that they posted it to 5,000 Usenet newsgroups, an act called *spamming*. Discussions about Celtic culture, Macintosh communications (where I first saw it), and Washington state politics were all interrupted, along with thousands of others completely apathetic about anything to do with immigration. Or at least they were apathetic until they were bludgeoned repeatedly with Canter & Siegel's post. All of a sudden, everyone cared a great deal about immigration, and sent 30,000 flame messages (more than 100 megabytes of text) to the offenders. That many messages was far more than Canter & Siegel's provider, Internet Direct, could handle, and their machine went down like a boxer on the wrong end of a knock-out punch.

The aftershocks keep coming, with Internet Direct suing Canter & Siegel for violating acceptable use policies (it seems that Canter & Siegel never signed the terms and conditions form) and for the detrimental effect the post had on business. In return Canter & Siegel counter-sued for loss of business, claiming some ludicrous percentage of the messages were requests for more information (though they refuse to provide any verifiable data). Needless to say, Internet Direct disabled their account immediately, and details about Canter & Siegel's history began to surface. They'd been kicked off of other providers for similar smaller-scale posts in the past, they'd been suspended from the bar in Florida in 1987 for conduct the Supreme Court of Florida deemed "contrary to honest," and so on. Canter & Siegel garnered a huge amount of press (most of it negative, but as the saying goes, "I don't care what you say about me as long as you spell my name right."). They even announced in a newspaper interview that they were setting up a company to provide services to other companies who wanted to flood Usenet with advertising, and they wrote a book (which has received awful reviews) about how to advertise on the Internet. That's a bit like Bonnie and Clyde writing a book about bank security.

The Canter & Siegel fiasco raises the question of how the Internet should be policed. In the past, and the present, any transgression has been dealt with much as it might have been in the perhaps-fictional view of the American Old West. Everyone takes justice into his own hands, and if a few innocents are hurt in the process, well, it was for the greater good. When Canter & Siegel spammed Usenet, thousands of people spammed them back.

This process is more commonly known as *mail bombing*. Mail bombs are generally small Unix programs (before you ask, I don't know of any for the Mac and don't condone their use in general) that simply send a user-configured number of messages (using a specified file as the message body) to a given address, potentially ensuring that none of the mail bomb messages come from real addresses. A better solution came from a Norwegian programmer, who created a spambot (his term, not mine) program that somehow figures out which newsgroups Canter & Siegel spammed (yes, it has happened again, although on a smaller scale each time) and bounces the spamming message back to them, along with a short note daring them to sue him, since he's in Norway.

Frontier justice sounds like great fun, especially when slimy lawyers are on the other end, but it raises some interesting issues. Mail bombing a machine doesn't affect just that machine—it affects many of the machines nearby on the Internet. In the case of a public machine like Internet Direct's **indirect.com**, it also hurts an innocent business and hundreds of innocent users who also use that machine. And, although the Internet as a whole can deal with the occasional mail bomb attack, if such things happened every day, they would seriously impair Internet communications. Such possibilities raise the specter of regulation, something that most Internet users disapprove of (though certain usage regulations are built into the service agreements of almost every Internet provider for liability reasons). So, will the government get involved and lay down the law about inappropriate Internet use? Probably not. The people who must do the regulation are the

providers themselves—there’s no way to prevent everyone from retaliating to such spam attacks as Canter and Siegel’s, so the best place to stop them is at the level of the providers. They can simply refuse to give problem users an account or remove accounts when abuse occurs. But the government itself? I certainly hope not.

I don’t believe that the Internet will ever be governed to a much greater extent than it now is (at least in the U.S.), simply because I don’t believe it’s feasible. How can you govern something that spans the globe or police something that carries gigabytes of data every day? The U.S. government could easily ban pornographic postings, say, but how does that affect someone from a different country? Or how does that affect U.S. users retrieving the pornographic images from another country? Remember, the Internet can just route around censorship. It’s all very confusing, and it will be some time (if ever) before the government understands all of the issues surrounding the Internet sufficiently to produce reasonable legislation. Of course, that begs the question of unreasonable legislation, but that’s always a fear.

The way the government as a whole currently views the Internet reminds me a bit of the joke about how to tell if you have an elephant in your fridge. The answer is by the footprints in the peanut butter—it’s the middle of the night, and the government is standing at the open door, yawning and blinking at those massive footprints. Luckily, different parts of the government are starting to wake up, which should help dispel the dangerous ignorance that has marked certain past government Internet actions.

For example, there was the Steve Jackson case, in which the Secret Service completely inappropriately confiscated the computer systems of a popular publisher of role-playing games. The damage award from that case enabled Steve Jackson Games to create an Internet provider called Illuminati Online (io.com). Perhaps the greater problem now with the government’s view of the Internet is that it seems more concerned with regulating occasional outrageous behavior than with using the power of the Internet to further the public good. Personally, I prefer my government to be more interested in helping than in regulating. Of course, then there are the people who would prefer that the government just stayed out of the way, but somehow I doubt that will happen any time soon.

Past and Future—Next, the Present

I’ve tried to give a glimpse of the history of the Internet, from its first stumbling steps as the military- and research-based ARPAnet to the swift NSFNET and NREN, along with the commercial providers of today. If nothing else, it’s worth noting that those who ignore history are condemned to repeat it, and by paying attention to the mistakes of the past, perhaps we can avoid making them again. The future will also bring new problems and new opportunities, but for the moment we can only speculate as to what those may be. But put all that out of your mind, because the next chapter takes you on a tour of the Internet of today.



Part II



Internet Foundations

So far, we've looked at the Internet in the abstract only, and it's important that you have an overview of the world you are entering. Like all things electronic, however, the Internet is terribly picky about the details; you must know exactly what to type and where to click. Moreover, unlike on your friendly local Macintosh, on the Internet real people see what you type, so I also talk about the social customs of the Internet, the manners and mores that everyone eventually learns. And, because I hope the Internet becomes something about which you talk with friends, I try to pass on some of the jargon and modes of speech.

Chapter 6 covers the basics of "Addressing & URLs" so you can figure out how email addresses and machine names are formed. In it, I also talk about Uniform Resources Locators, or URLs, which are used on the Internet to provide a coherent method of identifying Internet resources such as Web pages and files available via FTP. Chapter 7, "Email Basics," focuses on email usage and mailing lists, and chapter 8, "Usenet News Basics," is devoted exclusively to Usenet news. Chapter 9 describes "TCP/IP Internet Services," those being services that require a full Internet connection, such as Telnet, FTP, WAIS, Gopher, and the World Wide Web. Finally, chapter 10, "File Formats," discusses the main file formats you find on the Internet.

Keep in mind that this information is all background—I don't tell you the specific details of how to deal with programs on the Internet or anything like that, until Part III. Nevertheless, I feel that this is important background, so unless you've spent a fair amount of time on the Internet already, I recommend that you read through these chapters.

Text Styles

As a convention, I write all network addresses, whether they are machine names, full email addresses, or URLs, in this **monospaced font**. Note also that any punctuation following the address is not part of the address itself; instead, it's required by my seventh grade English teacher, who was adamant about ending clauses with commas and sentences with periods. Every now and then, I leave off a period when it confuses an address that ends a sentence, but the thought of her beet-red face (I'm sure she was very nice, but she reminded me of a lobster) looming over always makes me add that period. So remember that addresses *never* have any punctuation at the end.

Commands that you type exactly as written **look like this**; when there is a variable that you have to fill in, it *looks like this*. So, **TYPE *this*** means to type the word **TYPE**, followed by whatever is appropriate for *this*: your name, a file name, a directory name, a machine name, or whatever.

Finally, any text that shows up as though it scrolled by on a terminal window appears in its own monospaced font, line-by-line, much like the following lines:

```
To: The Reader <reader@ii.book.net>
Subject: Style conventions
From: Adam C. Engst <ace@tidbits.com>
```

I hope these conventions don't seem too onerous - it can be hard in a book to show precisely what the user sees, as opposed to what was or should be typed.

cheers ... -Adam

Chapter 6



Addressing & URLs

Before I can tell you about email, retrieving files via FTP, browsing the Web, or much of anything else, I must discuss how email addresses and machine names are formed, where they come from, and that sort of thing. Along with these details about email addresses and machine names, you must learn about URLs, or Uniform Resource Locators. URLs provide a coherent method of uniquely identifying resources on the Internet, ranging from Web pages to files available via FTP to WAIS sources. But let's start at the beginning...

Addressing

A rose may be a rose by any other name, but the same is not true of an Internet computer. All Internet computers think of each other in terms of numbers (not surprisingly), and all people think of them in terms of names (also not surprisingly). The Internet uses the *domain name system* to make sense of the millions of machines that make up the Internet. In terms of the numbers, each machine's address is composed of four numbers, each less than 256. People are generally bad about remembering more than the seven digits of a phone number, so the folks working in this field came up with a program called a *domain name server*. Domain name servers translate between the numeric addresses and the names; real people can remember and use the names while real computers can continue to refer to each other by number. That way, *everyone* is happy.

Note

Domain name servers, although generally part of the background technology that enables the Internet to work seamlessly, are tremendously important. Without them, very little on the Internet works these days.

Despite the fact that all Internet numeric addresses are sets of four numbers, the corresponding name can have between two and five sets of words. After five, it gets out of hand, so although it's possible, it's not generally done. For instance, one of the machines I use now is called **king.tidbits.com** (three words), and the machine I used at Cornell was called **cornella.cit.cornell.edu** (four words). The domain style addresses may look daunting, but in fact they are quite easy to work with, especially when you consider the numeric equivalents, such as **204.57.157.13** for **king.tidbits.com**. Each item in those addresses, separated by the periods, is called a *domain*, and in the following sections, you are going to look at them backward, or in terms of the largest domain to the smallest.

Note

A random aside for those of you who are students of classical rhetoric: The process of introducing topics A, B, and C, and then discussing them in the order C, B, and A is called chiasmus. This little known fact is entirely unrelated to the Internet, except that after the first edition of this book I took a lot of good-natured ribbing on the Internet about my classical education, so I figured I should at least pretend to know something about the topic.

Top-level Domains

In any machine name, the final word after the last dot is the *top-level domain*, and a limited number of them exist. Originally, and this shows the Internet's early Americo-centric view, six top-level domains indicated to what type of organization the machine belonged. Thus, we ended up with the following list:

- **com** = commercial
- **edu** = educational
- **org** = organization, usually nonprofit
- **mil** = military
- **net** = network
- **gov** = government

That setup was all fine and dandy for starters, but as the number of machines on the Internet began to grow at an amazing rate, a more all-encompassing solution became necessary. The new top-level domains are based on countries, so each country has its own two-letter domain. Thus, the United Kingdom's top-level domain is **uk**, Sweden's is **se**, Japan's is **jp**, Australia's is **au**, and so on. Every now and then another country comes on the Internet, and I see a domain code that totally throws me, as Iceland's **is** code did the first time.

Note

If you'd like to see the complete list of country codes, check out this URL:

<http://www.nw.com/zone/iso-country-codes>

The United States has this system, too; so, for example, The Well, a popular commercial service with links to the Internet, is **well.sf.ca.us**. Unfortunately, because so many sites already existed with the old domain names, it made no sense to change them. Thus, we have both types of top-level domain names here in the U.S., and you just have to live with it.

You may see a couple of other top-level domains on occasion, **bitnet** and **uucp**, such as in **listserv@bitnic.bitnet** or **ace@tidbits.uucp**. In both of these cases, the top-level domain indicates that the machine is on one of the alternative networks and may not exist directly on the Internet (otherwise, it would have a normal top-level domain such as **com** or **uk**). This setup isn't a big deal these days because so many machines exist on two networks that your email gets through just fine in most cases. In the past, though, few connections existed between the Internet and BITNET or Usenet, so getting mail through one of the existing gateways was more difficult. Keep in mind that because a machine whose name ends with **bitnet** or **uucp** is not usually on the Internet, you cannot use Telnet or FTP with it.

Many machine names are as simple as it gets: a machine name and a top-level domain. Others are more complex because of additional domains in the middle. Think of an address such as **cornella.cit.cornell.edu** as one of those nested Russian dolls (see figure 6.1). The outermost doll is the top-level domain, the next few dolls are the mid-level domains, and, if you go all the way in, the final doll is the userid (which I'll explain soon enough).

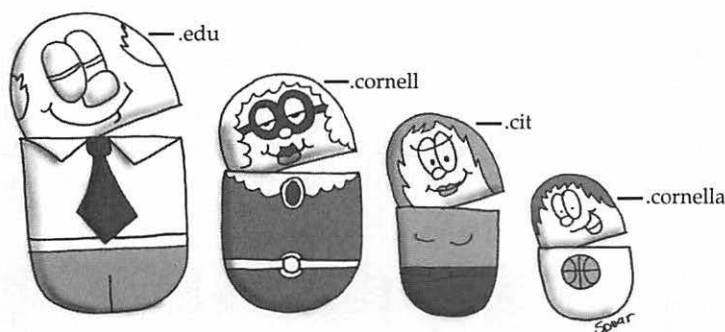


Figure 6.1 The Russian doll approach to Internet addresses.

Mid-level Domains

What do these mid-level domains represent? It's hard to say precisely, because the answer can vary a bit. The machine I used at Cornell, known as **cornella.cit.cornell.edu**, represents one way the mid-level domains have been handled. The machine name is **cornella**, and the top-level domain is **edu**, because Cornell claims all those undergraduates are there to get an education. The **cit** after **cornella** is the department, Cornell Information Technologies, that runs the machine known as **cornella**. The next part, **cornell**, is obvious; it's the name of the overall organization to which CIT belongs. So, for this machine anyway, the hierarchy of dolls is, in order, machine name, department name, organization name, and organization type.

This is similar to how my system is set up now, since I control the **tidbits.com** domain, and each of my Macs has a name within that domain. So, for instance, my desktop Mac is called **penguin.tidbits.com**, and my server is **king.tidbits.com**.

In the machine name for The Well, **well.sf.ca.us**, you see a geographic use of mid-level domains. In this case, **well** is the machine name, **sf** is the city name (San Francisco), **ca** is the state name (California), and **us** is the country code for the United States.

Mid-level domains spread the work around. Obviously, the Internet can't have machines with the same name; otherwise, chaos would erupt. But because the domain name system allows for mid-level domains, the administrators for those mid-level domains must make sure that everyone below them stays unique. In other words, I could actually name my machine **cornella.tidbits.com** because that name is completely different from **cornella.cit.cornell.edu** (though why I'd want to, I don't know). And, if they wanted, the administrators at CIT could put a new machine on the net and call it **tidbits.cit.cornell.edu** without any trouble, for the same reason. More importantly, the administrators don't need to bother anyone else if they want to make that change. They control the **cit** domain, and as long as all the machines within that domain have unique names, there aren't any problems. Of course, someone has to watch the top-level

domains because it's all too likely that two people may want **tidbits.com** as a domain (but I've already got it, so they can't have it). That task is handled by the Internet Network Information Center, or InterNIC. As a user, you shouldn't have to worry about naming problems, because everyone should have a system administrator who knows who to talk to, and you need the cooperation of your provider anyway—you can't set up a domain on your own.

There is yet another way to handle the mid-level domains, this time in terms of intermediate computers. Before I got my current address, I had a connection from a machine called **halcyon**, whose full name was **halcyon.com**. My machine name was **tidbits.halcyon.com**. In this case, **tidbits** was my machine name, **halcyon** was the machine through which all of my mail was routed, and **com** indicated that the connection was through a commercial organization. I realize that this example is a bit confusing, but I mention it because it's one way that you can pretend to have an Internet address when you really have only a UUCP connection (a different sort of connection that transfers only email and news). All my mail and news came in via UUCP through **halcyon**, so by including **halcyon** in my address, I created an Internet-style address.

The other way of pretending that a UUCP connection is a real Internet connection for address purposes, is to have your host set up an *MX record* (where MX stands for Mail Exchange). An MX record is a pointer on several true Internet machines to your site.

Machine Name

The next part in the full domain name is the machine itself; for example, in the name **penguin.tidbits.com**, **penguin** is the name of my machine. In my case, the machine is a Macintosh 660AV, but people use all sorts of machines, and because the system administrators often are a punchy, overworked lot, they tend to give machines silly names. Large organizations with more centralized control lean more toward thoroughly boring names, like the machine at Cornell, which was called **cornella** (as opposed to **cornellic** and **cornellid** and **cornellf**).

Note

*For those who are wondering, the naming scheme I use is based on the names of species of penguins. Also, if you're wondering why you can send email to **ace@tidbits.com** if my machine is really called **penguin.tidbits.com**, it's because of the magic of the domain name system. Since most people like shorter addresses, it's common to map the shorter domain name, **tidbits.com**, to point to the server that handles mail specifically, **king.tidbits.com** in my case. Then, I set Eudora to look for mail on **king.tidbits.com** and everything works swimmingly. These are the sort of machinations that Internet providers continually deal with. Luckily, you as the user can usually ignore them.*

One of the reasons for boring names is that in the early days, machines on BITNET had to have names with between six and eight characters. Coming up with a meaningful unique name within that restriction became increasingly difficult. Usenet doesn't put a limit on the length of names, but it requires that the first six characters be unique. Currently, the Internet allows the second level domain to be up to 24 characters, and the third level domain can be up to 72 characters. In no case can the full domain name go over 256 characters, however.

If you remember that machines often exist on the Internet as well as on one of these other networks, thereby blurring the distinctions, you'll see the problem. The limitations of Internet machine names are less rigid, so alternative connections dictate what names are acceptable.

Often, special services keep their names even when they move to different machines or even different organizations. Because of this situation, a machine that runs a service may have two names, one that goes with the machine normally and one that points solely at that service. For instance, the anonymous FTP site that I use to store all the software I talk about in this book is called **ftp.tidbits.com**. But in fact, it runs on a machine called **ftp.halcyon.com**, and I could move it to any other machine while still retaining the **ftp.tidbits.com** name. This situation is not a big deal one way or another.

To summarize, you can have multiple domains in a machine name, and the further you go to the right, the more general they become, often ending in the country code. Conversely, the further you go to the left, the more specific the domains become, ending in the machine name because it's the most specific.

But what about email addresses, which have userids? They're even more specific than machine names, because you can have many userids on a single machine.

Userid

Now that you've looked at the machine name, you can move on to the *userid* or *username*, which identifies a specific user on a machine. Both terms are equally correct (with two exceptions—the commercial online service GENie and the FirstClass BBS software both treat userids and usernames separately) and commonly used. If you set up your own machine, or work with a sufficiently flexible provider, you can choose your own username. Choosing your own name is good because then your correspondents can more easily remember your address, assuming of course that you choose a userid that makes sense and is easy to type. If I made my address **ferdinand-the-bull@tidbits.com**, people who typed the address slightly wrong and had their mail bounced back to them would become irritated at me.

Unlike Macintosh filenames (and America Online and eWorld userids), Internet userids cannot have spaces in them, so convention dictates that you replace any potential spaces with underscores, dashes, or dots, or omit them entirely. Other reasonable userids that I

could use (but don't) include `adam_engst@tidbits.com` or `adam-engst@tidbits.com` or `adam.engst@tidbits.com` or `adamengst@tidbits.com`. However, all of these names are more difficult to type than `ace@tidbits.com`, and because I have good initials, I stick with them.

Unfortunately, there are a limited number of possible userids, especially at a large site. So Cornell, for instance, with its thousands of students and staff, has opted for a system of using initials plus one or more digits (because initials aren't all that unique, either—in fact, I once asked for my initials as a userid on one of Cornell's mainframes and was told that ACE was a reserved word in that machine's operating system, though no one could tell me what it was reserved for).

Microsoft uses yet a different scheme: first name and last initial (using more than one initial to keep the userids unique). As Microsoft has grown, common names such as David have been used up, so the company has started other schemes such as first initial and last name. Why am I telling you this? Because knowing an organization's scheme can prove useful at times if you're trying to figure out how to send mail to someone at that organization, and so that I can note a societal quirk. At places like Microsoft where people use email so heavily, many folks refer to each other by email names exclusively. When my wife, Tonya, worked at Microsoft, she had a problem with her username, **tonyae** (first name and last initial) because it looked more like TonyAe than TonyaE to most people.

The real problem with assigned userids comes when the scheme is ludicrously random. Some universities work student ID numbers into the userid, for instance, and CompuServe userids are mere strings of digits like 72511,306. I believe the scheme has something to do with octal numbers or some such technoweenie hoo-hah. I don't speak octal or septal, or any such nonsense, and as a result, I can never remember CompuServe userids.

Remember that email addresses point at an individual, but when you're using services such as Telnet or FTP, no individual is involved. You simply want to connect to that machine, and you have to connect *sans* userid. This restriction may seem obvious, but it often trips people up until they get used to it. For example, it seems that you could just FTP to `anonymous@space.alien.com`. The system doesn't work that way, though, and you FTP to `space.alien.com`, and once there, log in as `anonymous`. More about FTP in later chapters.

Punctuation

Enough about userids. What about all this punctuation? Better known as Shift-2 (on U.S. keyboards anyway), the @ symbol came into use, I imagine, because it's a single character that generally means "at" in traditional usage. The @ symbol is generally universal for Internet email, but not all types of networks have always used it. For instance, some BITNET machines once required you to spell out the word, as in the command **TELL LISTSERV AT BITNIC HELP**. Luckily, almost everything uses the @ symbol with no spaces these days, which reduces four characters to one, and probably has saved untold person-hours worth of typing over the years.

On some European keyboards there is no @ key, but you can still type the character by pressing ⌘-Option-1.

As long as you're learning about special characters, look at the dot. It is, of course, the period character on the keyboard, and it serves to separate the domains in the address. For various reasons unknown to me, the periods have become universally known as dots in the context of addresses. When you tell someone your email address over the phone, you say (or rather I'd say because it's my address), "My email address is ace at tidbits dot com." The other person must know that "at" equals the @ symbol and that "dot" equals the period. If he's unsure, explain yourself.

Alternative Addresses

You may see two other styles of addressing mail on the Internet, both of which work for sites that aren't actually on the Internet itself. The first, and older, of the two is called *bang* addressing. It was born in the early days when there were relatively few machines using UUCP. Not every machine knew how to reach every other machine, so the trick was to get the mail out to a machine that knew about a machine that knew about a machine that knew about your machine. Talk about a friend of a friend! So, you could once have sent email to an address that looked like **uunet!nwnexus!caladan!tidbits!ace**. This address would have sent the mail from **uunet** to **nwnexus** to **caladan** to **tidbits** and finally to my userid on **tidbits**. This approach assumes that your machine knows about the machine **uunet** (run by the commercial provider UUNET) and that all of the machines in the middle are up and running. All the exclamation points are called "bangs," appropriately enough, I suppose. On the whole, this style of addressing is slow and unreliable these days, but if you use a machine that speaks UUCP, you can occasionally use it to your advantage. For instance, every now and then, I try to send email to a machine that my UUCP host, **nwnexus.wa.com**, for some reason can't reach. By bang-routing the mail appropriately, I can make another Internet machine try to send the mail out, sometimes with greater success.

The other sort of special addressing is another way to get around the fact that your machine, or even your network, isn't connected to the Internet as such. In this case, you must provide two addresses: one to get to the machine that feeds your machine, and one to get to your machine. The problem here is that Internet addresses cannot have more than one @ symbol in them. You can replace the first @ symbol with a % symbol, and the mailers then try to translate the address properly. My old address, **ace@tidbits.halcyon.com**, also could have been **ace%tidbits.uucp@halcyon.com**. These tricks are ugly and awkward, but sometimes necessary. Luckily, as the Internet grows and standardizes, you need fewer and fewer of these addressing tricks.

Enough on machine names and email addresses, then. If you keep the previous discussions in mind when you're using the Internet, you shouldn't be confused by any address you see. And if you are confused, perhaps that address is seriously malformed. I've seen it happen before.

URLs

Before I talk about any of the various TCP/IP-based Internet services, I want to explain URLs, or Uniform Resource Locators. These constitute the most common and efficient method of telling people about resources available via FTP, the World Wide Web, and other Internet services. URLs have become so popular that the Library of Congress has even added a subfield for them when it catalogs electronic resources.

Note *URL generally stands for Uniform Resource Locator, although some people switch “uniform” for “universal.” Despite what I’ve heard from one source, I have never heard anyone pronounce URL as “earl;” instead, everyone I’ve talked to, including one person from CERN who helped develop the World Wide Web, spells out the letters.*

What are URLs?

A URL uniquely specifies the location of something on the Internet, using three main bits of information that you need in order to access any given object. First is the URL scheme, or the type of server making the object available, be it an FTP, Gopher, or World Wide Web server. Second, comes the address of the resource. Third and finally, there’s the full pathname or identifier for the object, if necessary.

Part II
Addressing
& URLs

Note *Don’t worry if I talk about Internet services that you haven’t read about in detail yet. That’s what the next few chapters are for, but I wanted to explain the way that people (including me in this book) provide pointers to specific resources available via the various services like FTP, Gopher, and the World Wide Web.*

This description is slightly oversimplified, but the point I want to make is that URLs are an attempt to provide a consistent way to reference objects on the Internet. I say “objects” because you can specify URLs not only for files and Web pages, but also for stranger things, such as email addresses, Telnet sessions, and Usenet news postings.

Table 6.1 shows the main URL schemes that you’re likely to see.

Table 6.1
Common URL Schemes

<i>Scheme</i>	<i>Internet Protocol</i>	<i>Sample Client</i>
ftp	File Transfer Protocol	Anarchie
gopher	Gopher protocol	TurboGopher
http	HyperText Transfer Protocol	MacWeb
mailto	Simple Mail Transport Protocol	Eudora
news	Net News Transport Protocol	NewsWatcher
wais	Wide Area Information Servers	MacWAIS

URL Construction

If you see a URL that starts with **ftp**, you know that the file specified in the rest of the URL is available via FTP, which means that you could use FTP under Unix, FTP via email, or a MacTCP-based FTP client such as **Anarchie** to retrieve it. If the URL starts with **gopher**, use **TurboGopher** or another Gopher client. If it starts with **http**, use **MacWeb**, **NCSA Mosaic**, or **Netscape** or some other Web browser. And, finally, if a URL starts with **wais**, you can use **MacWAIS** or another WAIS client.

Note

You can use a Web browser to access most of the URL schemes in Table 6.1, although Web browsers are not necessarily ideal for anything but information on the World Wide Web itself. Web browsers work pretty well for accessing files on Gopher servers and via gateways to WAIS databases, but FTP via a Web browser is clumsy (and may fail entirely with certain types of files, such as self-extracting archives). Similarly, although it's handy to use mailto URLs to send mail, I dislike doing so because then I don't have a record of my outgoing mail, as I do when I send mail from Eudora. And, no Web browser stands up to NewsWatcher in terms of news capabilities.

After the URL scheme comes a colon (:), which delimits the scheme from what comes next. If two slashes (//) come next, they indicate that a machine name in the format of an IP address will follow, such as with **http://www.apple.com/** or **ftp://ftp.info.apple.com/**. However, if the URL points at an address in some other format, such as an email address like **mailto:president@whitehouse.gov**, the slashes aren't appropriate and don't appear.

In some rare circumstances, you may need to use a username and password in an FTP URL as well. A URL with a username and password might look like this:

```
ftp://username:password@domain.name/pub/
```

The last part of the URL is the specific information that you're looking for, be it an email address or more commonly, the path to the directory of the file that you desire. Directory names are separated from the machine name by a slash (/). You may not have to specify the path with some URLs if you're only connecting to the top level of the site.

So, for instance, let's dissect a URL that points at the Product Support page on Apple's Web server:

```
http://www.apple.com/documents/productsupport.html
```

First off, the **http** part tells us that we should use a Web browser to access this URL. Then, **www.apple.com** is the name of the host machine that's running the Web server. The next part, **/documents/productsupport.html**, is the full path to the file the Web browser shows us, so **/documents** is a directory, and **productsupport.html** is the actual file inside the **/documents** directory.

If an FTP or Gopher URL ends with a slash, that always means it points at a directory and not a file. If it doesn't end with a slash, it may or may not point at a directory. If it's not obvious from the last part of the path, there's no good way of telling until you go there. Thus, this URL points at a directory and will return the directory listing of the files there:

```
ftp://ftp.tidbits.com/pub/tidbits/
```

However, this URL points directly at a file:

```
ftp://ftp.tidbits.com/pub/tidbits/issues/1990/TidBITS#001_16-Apr-90.etx
```

Because most Web servers enable the creation of a default file that serves in the absence of a specific file in the URL, it's usually less important for Web users to realize whether or not they're specifying a file or a directory. In other words,

```
http://www.tidbits.com/tidbits/index.html
```

points at a file, but the Web server running on that machine will display the same file (because it's the default), if you simply used this URL:

```
http://www.tidbits.com/tidbits/
```

Using URLs

All of these details aside, how do you use URLs? Your mileage may vary, but I use them in three basic ways. First, if I see them in email or in a Usenet posting, I often copy and paste the host part into Anarchie (if they are FTP URLs), or I paste the whole thing into MacWeb or Netscape (if any other scheme). That's the easiest way to retrieve a file or connect to a site if you have a MacTCP-based Internet connection.

Note

Actually, thanks to some slick programming, all I'd really do is Command-click on the URL in NewsWatcher, say, and it would automatically transfer that URL to the appropriate client program, Anarchie, TurboGopher, MacWeb, or whatever.

Second, if for some reason I don't want to use MacWeb or Netscape (I far prefer Anarchie for FTP, for instance), sometimes I manually dissect the URL, as we did with the Product Support page on the Apple Web server, to figure out which program to use and where to go. This method takes more work, but sometimes pays off in the end. (You *can* put a screw in the wall with a hammer, but it's not the best tool for the job.)

Third and finally, whenever I want to point people to a specific Internet resource or file available for anonymous FTP, I give them a URL. URLs are unambiguous, and although a bit ugly in running text, easier to use than attempting to spell out what they mean. Consider the example below:

```
ftp://ftp.tidbits.com/pub/tidbits/issues/1995/TidBITS#261_30-Jan-95.etx
```

To verbally explain the same information contained in that URL, I would have to say something like: "Using an FTP client program, connect to the anonymous FTP site **ftp.tidbits.com**. Change directories into the **/pub/tidbits/issues/1995/** directory, and once you're there, retrieve the file **TidBITS#261/30-Jan-95.etx**." A single URL enables me to avoid such convoluted (and boring) language; and frankly, URLs are in such common use on the Internet, you may as well get used to seeing them right now.

Note

URLs sometimes have to break between two lines in publications. If you see a two-line URL that doesn't look quite right, stick the two lines back together when you're typing or pasting it into a Web browser, perhaps without a hyphen that might have been introduced in the production process.

So, from now on, whenever I mention a file available via FTP or a Web site, I'll use a URL. If you try to retrieve a file or connect to a Web site and are unsuccessful, chances are either you've typed the URL slightly wrong, or the file or server no longer exists. It's

extremely likely that many of the files I give URLs for will have been updated by the time you read this, so the file name at the end of the URL may have changed.

So if a URL doesn't work, and this is a general piece of good advice, try removing the file name from the last part of the URL and look in the directory that the original file lived in for the updated file. If all else fails, you can remove everything after the machine name and work your way down to the file you are after.

If, after all this, you'd like to learn more about the technical details behind the URL specifications, check out the following:

<http://info.cern.ch/hypertext/WWW/Addressing/URL/Overview.html>

Weird Characters

There is one rather messy part to URLs that you don't usually have to deal with, but that comes up on occasion, most commonly in relation to Gopher URLs. There are certain characters which cannot appear in certain parts of a URL, including spaces. And if one of those characters would appear, it's replaced with what's called an *escape code*, consisting of a percent symbol and the hexadecimal number corresponding to that character.

The reason this comes up most often in relation to Gopher URLs is that Gopher allows extremely long titles for files and directories, and allows pretty much any character within them, including spaces, slashes, question marks, and so on. So a Gopher URL may look a bit like this:

<gopher://gopher.tc.umn.edu/11/Information%20About%20Gopher>

Notice all the %20 escape codes that stand in for what are spaces on the real Gopher menu title.

For the most part, you don't have to worry about the way the spaces and other characters (see Table 6.2 for a list of some common ones that will show up in a URL as escape codes) are translated—I just wanted to show you that this sort of thing happens so you won't be confused the first time you see a URL with all sorts of what seem like garbage characters in it.

Table 6.2

Some Reserved Characters in URLs

Character	Escape Code Replacement
=	%3D
;	%3B

continues

Table 6.2

continued

<i>Character</i>	<i>Escape Code Replacement</i>
/	%2F
#	%23
?	%3F
:	%3A
space	%20
~	%7E

A Self-Addressed Internet...

I believe that I promised early on in this book that there would be no quiz, but if I were going to break my promise, this is probably the chapter I'd do it in. You cannot get around on the Internet unless you understand how machine names and email addresses are put together. And, as the World Wide Web continues on its steamroller path to become the most popular of Internet services, a working knowledge of URLs, no matter how ugly they may seem to you now, is absolutely essential if you're to understand where you're going and what you're seeing.

Chapter 7



Email Basics

Electronic mail is the most pervasive application on the Internet, and for good reason. What better way to communicate with so many people so quickly? But to use and understand email properly, I must show you how it's constructed, the relevant social mores and pitfalls, and how you can use it.

Email Construction

What makes up an email message? Most messages have two important parts, with a third part that doesn't have to appear. The first two parts are the *header* and the *body* of the message, and the third, non-essential part, is the *signature*. For simplicity's sake, let's work backward.

Signature

Signatures are just about what you'd expect—some text that goes at the bottom of every message you send; many email programs, including Eudora, which I've included on the *Internet Interactive* CD, provide a facility for creating signatures. Most people include their names (real or pretend) in their signatures; it's considered good form to include your preferred email address in your signature as well, just in case the address in the header isn't useful for some reason or another. After you get past the basics of name and email address, however, you can put anything you like in your signature. Many people lean toward clever quotations or manage to express some sporting partisanship of their favorite team, usually with an erudite "Go Weasels '95" or some such. (It's hard to grunt in ASCII.)

paragraphs to insert a blank line between them; that additional white space makes email messages much easier to read. Nothing is harder to read than page after page of unbroken text.

Actually, something *is* worse than unbroken text, and that's page after page of unbroken text in capital letters. **DON'T USE ALL CAPS BECAUSE IT LOOKS LIKE YOU'RE SHOUTING!** No one uses all capital letters for long because everyone hates reading it and will tell you, nicely the first time, to stop.

I suppose now is a good time to talk about manners in terms of the sorts of things you should consider when writing email. Email differs from normal mail in many ways. Think of the difference between a short note to your mother, a memo at work, and a formal business letter. Most email falls somewhere between the short note and the memo, and seldom do you ever see an email message with the formality and rigidity of a business letter. Although I'm giving this information in the context of email, it applies equally as well to postings on Usenet; so if you like, reread this section, substituting posting for email everywhere.

How do you start these messages? In many ways, email acts as the great equalizer. Most of the time, you know someone's name and email address when you send email to him, nothing more. **joeschmoe@alien.com** could be a janitor, a summer intern, or the president of a Fortune 500 firm. Similarly, any address ending in **edu** can link to a student, some member of the staff, a world-renowned professor of underwater basket weaving, or the president of the university. You have no way of knowing, unless that fact somehow comes up in conversation.

Most people react to this lack of context by treating everyone with the same level of polite, but informal, respect. Seldom do people use their titles, so equally seldom do correspondents use those titles in email. Everyone is on a first-name basis. I once took a class with the astronomer and science advocate Carl Sagan while I was at Cornell, and the first day of class, an awed undergraduate (but braver than the rest of us) asked, "How should we address you, Dr. Sagan?" He replied, "You can call me Mr. Sagan, Professor Sagan, Dr. Sagan, or Herr Doktor Professor Sagan," he paused, "or you can call me Carl." Carl it was then, and the class benefited greatly from that level of informality, just as the Internet does.

In light of this knowledge, when I started using email I thought about the differences between email and paper mail (hereafter called by its true name in the Internet community, snail mail). The standard salutation of "Dear" sounds inappropriately formal and stilted to my ears (apologies to Miss Manners). Since email more closely resembles spoken communication than written, I opted for the less formal and more colloquial "Hi," which has served me well. Some people forego the salutation completely, relying solely on the first name, but that approach feels abrupt to me, as if someone called me on the phone and stated my name without a "Hello" or so much as a questioning tone. Do what you like, though; no one has laid down rules on this matter.

What you say in the letter itself deserves more thought, however. Because email is so quick and it's so easy to respond without thinking, many people often reply hastily and less politely than they would had they taken a moment to consider. Remember, you want to achieve a certain effect with an email message, just as you do with any form of communication. If you simply whack your first thoughts into a message, it probably won't properly convey your true feelings. If you want information from someone, phrasing your request politely only increases your chances of getting that information, and if you wish to comment on someone else's words, doing so in a reasoned and level-headed manner ensures that that person won't immediately consider you a serious jerk.

You also must remember that informal though email may be, it lacks most of the nonverbal parts of communication that we seldom consider in normal speech. All inflection, body language, and facial expressions disappear, and it doesn't help one whit if you make them while composing the letter. Email is ASCII text only, and only two ways exist to convey inflections such as sarcasm or irony that would be obvious in spoken conversation. First, polish your writing skills. There is no substitute for clear and coherent writing. Many people find writing difficult, but I recommend that you don't think of composing email as writing, but as speaking to someone who sees your words but cannot see or hear you. Most people who claim they can't write have little trouble making themselves understood when speaking.

Second, utilize *smileys*, or as they are sometimes known, *emoticons*. Smileys are strings of punctuation characters meant to be viewed by tilting your head (which is usually easier than tilting your monitor) to the side. (If they still look wrong, try tilting your head to the *other* side.) People have come up with literally hundreds of different smileys, and you can find lists containing them on the Internet. Seth Godin has even compiled many of them into a book, *The Smiley Dictionary*, published by Peachpit Press (and there is at least one other book, published by O'Reilly & Associates, on the same somewhat silly topic). I take the view that only two, or maybe three, smileys are at all useful in normal email. The first is the happy face :-), which implies that what you just said was meant as humor or at least shouldn't be taken too seriously. I often use it to imply that I would have said that bit with a smile. A variant of the happy face uses the semicolon instead of the colon ;-) and (because of the wink) implies that the preceding sentence was somewhat sarcastic or ironic. Finally, the frowning face :-(implies that you aren't happy about whatever you just said.

I use smileys relatively heavily in email, where I don't have time to craft each letter as carefully as I would like. I miss not being able to use them (I could, but no one would understand) in snail mail occasionally, and I actively try to avoid using them in *TidBITS*, favoring instead words that convey my feelings without the smiley crutch. When in

doubt, use smileys. If I say in email, “Well, that was a stupid thing to do,” the message is much more offensive than if I say, “Well, that was a stupid thing to do. :-)” Believe me, it is.

I may have given the impression that the Internet is this utopia where everyone always behaves nicely and ne’er is heard a discouraging word. Unfortunately, that’s not so, and in reality you see plenty of *flaming* on the nets. Flaming happens when, in a PC discussion list, you innocently mention that you like your Macintosh, and 17 people immediately jump on you in email and pummel you within an inch of your electronic life for saying something so obviously stupid and incorrect when everyone knows that only weenies, wimps, and little wusses use those toy Macintosh computers, which are good only for paperweights—and expensive paperweights at that, because you can buy three completely configured, top-of-the-line Pentium-based PCs for the same price as a used Macintosh Classic—without a hard drive. And by the way, did I mention that your mother wears combat boots and your father wears ballet slippers? :-)

The preceding paragraph is flaming (except for the smiley, which I threw in to indicate that I was kidding about your parents’ footwear), and if you must respond to an inflammatory message, which I don’t recommend, do it in email. No one else wants to read your flames. Think before you lower yourself to flaming; it never solves anything. I have found in almost every case that replying calmly and clearly embarrasses anyone (assuming that person is normal and rational, which is not always a good assumption) so thoroughly that she immediately apologizes for being such a jerk. And yes, I know how hard it is not to just tee off on someone. Restrain yourself and rest assured that everyone who sees your restraint will think more highly of you for it.

Note

My favorite method of responding to long and vitriolic flames is to send back a single-line message reading, “You may be right. -Engst.” I heard about this technique in an interview with Tom Brokaw, I think, and it works extremely well, confusing the recipient to no end and generally putting a stop to the flaming.

Often, people flame companies or large organizations that are doing stupid things. Various governments are favorite, though slow-moving and not very challenging, targets. This sort of flaming is more acceptable, although you may start a *flame war* if other people don’t share your opinions on some major topic, such as whether the Mac is better than Windows. As a spectator, you may enjoy watching the occasional flame war, as I do, but again, they never solve anything, and they waste a huge amount of *bandwidth* (which is composed of transmission time, people time, and disk storage throughout the world).

Note

Actually, I've decided that in some respects a certain amount of flaming can be positive, because there are only three ways of ending an argument on the Internet. Agree to disagree, win your opponent over to your side, or stop from exhaustion. In no case does anyone get knifed or shot, and if participating in a flame war lets someone blow off some steam, that's better than their going home and abusing their children. Everything is relative.

Keep in mind that no matter what you say, it may not be private. Always assume that gobs of people can and do read every message you send. These people include your coworkers, your system administrator, system administrators on other machines through which your email travels, random pimply faced fools who like poking around in other people's email, and last, but certainly not least, the government, probably in the form of the CIA, FBI, Secret Service, or National Security Agency. I realize this sounds alarming, and it is most certainly not completely true, but the possibility exists for all of these people to read your email.

In reality, email carries significant privacy, but because you have no guarantee of that privacy, you should stay aware of what you're saying. This suggestion is especially true if you use email at work, where you could lose your job over ill-considered remarks in email. It's always a good idea to check on your employer's policy about email privacy.

Note

There have been a number of court cases regarding ownership of email (Does it belong to you? Does it belong to your employer?) at some large companies like Epson and Borland. Because it may come down to a matter of their lawyers being meaner than your lawyers, don't push it.

This lack of privacy carries over to mailing lists and Usenet news (where you want people to read your messages, but you may not want the government to keep tabs on your postings). In fact, some people have gone so far as to include inflammatory keywords in otherwise innocuous postings, just to trip up the rumored government computers scanning for terrorists, assassins, space aliens, nudists, vegetarians, people who like broccoli, and other possible undesirables.

I almost forgot about *attachments*. Many people like to send each other files in email, and although you can do this simply by encoding the file as BinHex or uucode (I'll talk about these file formats in a few chapters) and pasting it into the body of the message, modern email programs instead enable you to merely attach the file to the message with a specific command.

That's all fine and nice if your recipient also uses an email program that knows how to deal with the attachment, but if not, your friend sees the file, usually encoded in BinHex or uuencode, at the end of the message in the body (but before the signature). Large email files can be a pain to deal with unless your email program supports attachments.

Header

Okay, I admit it; I've been avoiding talking about the header so far. I did so because the message header generally looks like a lot of gobbledygook to the novice user, and in fact, it should. The header exists for the computers, not for the users, and you're lucky that you can read it as well as you can. In some programs you can see an abbreviated header, which is good, and in some cases you can ignore the header altogether, which can be a little dangerous because it may not be clear who receives a reply to that message.

As much as the header is technoweenie information that exists primarily for the computers to route mail to you, I recommend that you choose an abbreviated header display if you have one. An abbreviated header shows you information that can be useful, such as who sent the email to you, when it was sent, what the subject of the message is, and to whom it was sent (not always only you—it's easy to send the same piece of email to multiple people).

Take a look at a typical header, culled just now from one of my archived pieces of email (see figure 7.1).

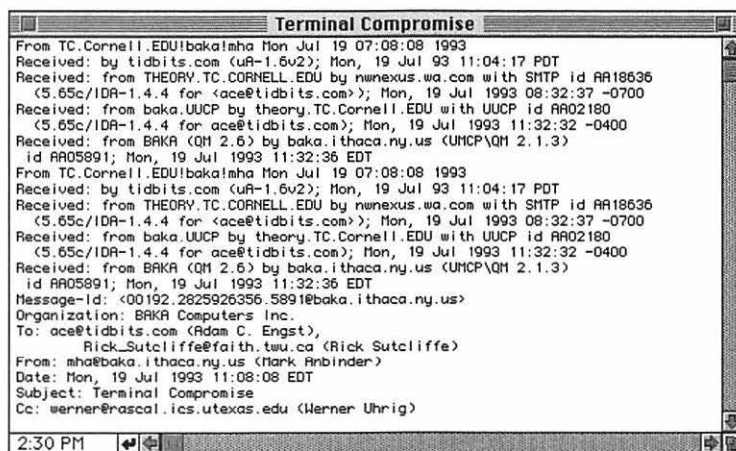


Figure 7.1 A sample header.

Let's take a spin through all the different parts of the header, explaining each one along the way and starting with the glop at the top.

```
From TC.Cornell.EDU!baka!mha Mon Jul 19 07:08:08 1993
Received: by tidbits.com (uA-1.6v2); Mon, 19 Jul 93 11:04:17 PDT
Received: from THEORY.TC.CORNELL.EDU by nwnexus.wa.com with SMTP id AA18636
(5.65c/IDA-1.4.4 for <ace@tidbits.com>); Mon, 19 Jul 1993 08:32:37 -0700
Received: from baka.UUCP by theory.TC.Cornell.EDU with UUCP id AA02180
(5.65c/IDA-1.4.4 for ace@tidbits.com); Mon, 19 Jul 1993 11:32:32 -0400
Received: from BAKA (QM 2.6) by baka.ithaca.ny.us (UMCP\QM 2.1.3)
id AA05891; Mon, 19 Jul 1993 11:32:36 EDT
```

The preceding lines are merely routing information that tell you where a message went and when it arrived there. You have to read it backwards to follow the flow, so this message traveled from the QuickMail server at BAKA Computers to the UMCP Bridge (**baka.ithaca.ny.us**), which acts as a gateway for QuickMail messages destined for the Internet. From there it traveled to Cornell's Theory Center, **theory.tc.cornell.edu**, and from the Theory Center, the message bounced almost instantly to **nwnexus.wa.com**, which is my Internet host, and then several hours later to my machine, **tidbits.com**.

You generally can ignore this part of the header, although it can be fun to see where your message went at times. If your message *bounces*—that is, if it fails to go through for some reason and comes back to you—looking at this part in the header helps you determine how far it got and which machine didn't like it. More about how to handle bounces in a bit.

Message-Id: <00192.2825926356.5891@baka.ithaca.ny.us>

The Message-Id line uniquely identifies each message. It's generally of no use at all, and although it looks like an email address, it's not.

Note

Only once have I found the Message-Id information useful. For some reason, one of my hosts was duplicating some files that went out, and often the only difference between the messages was that at a certain point they started having different ids. Unfortunately, I never figured out how to solve the problem; I just switched to another host.

Organization: BAKA Computers Inc.

The Organization line identifies the organization of the sender, as you may suspect. Individuals often have a good time with this line because they don't have real organizations to put down and can thus include fake organizations like "Our Lady of the Vacant Lot Enterprises."

To: ace@tidbits.com (Adam C. Engst),
Rick_Sutcliffe@faith.twu.ca (Rick Sutcliffe)

The To line can have one or more entries, and it specifies, reasonably enough, who the mail was sent to. The recipient may not be you because you might be the person mentioned in the Cc line or even the Bcc line (which you don't see because Bcc stands for Blind Carbon Copy). Most of the time you see a name before or after the email address, but it's not mandatory.

From: mha@baka.ithaca.ny.us (Mark Anbinder)

The From line indicates who sent the email and is self-explanatory.

Date: Mon, 19 Jul 1993 11:08:08 EDT

The Date line lists the date that the email was sent originally, not the time you received it or read it, and should usually indicate the time zone in which the sender lives. Even then I find it difficult to keep track of what time it is in other countries. Do you know the local time in Turkey right now? Some messages use a number, either positive or negative, and the acronym GMT, which stands for Greenwich Mean Time. Unfortunately, this use requires that you know what time it is in Greenwich, England, and that you know how your local daylight savings time is involved. Date lines usually just confuse me.

Note

The best way to determine what time it is in another part of the world is to open the Map control panel, click the appropriate spot on the map of the world, and look at the time display. Finding the appropriate spot is up to you. I hope you paid attention in geography class in grade school.

Subject: Terminal Compromise

The Subject line should give a clear and concise description of the contents of the email message. In practice, this description often isn't true, especially after a discussion proceeds, changing topics occasionally, with everyone using the reply function to keep the Subject the same. After a while the Subject line bears no resemblance to the contents of the message, at which point it's time to change the line.

Cc: werner@rascal.ics.utexas.edu (Werner Uhrig)

The Cc line lists all the people who received copies of the message. There is no functional difference from being on the To line or the Cc line, but in theory if you receive only a copy, the message shouldn't concern you as much. In practice I notice little difference.

An abbreviated header probably just shows these five lines and avoids displaying the routing information at the top of the header.

You also may see other lines in the header that identify which program mailed the message, to whom the recipients should reply, the type of data included in the message, how the data is encoded, and that sort of thing. In general, you don't have to worry about anything in the header very much, but it's worth taking a look every now and then to see if you can tell what's going on in there.

Bounces

In a perfect world, all email would get through to its destination quickly and reliably. But just as with snail mail, which can take one to five days to appear, and which sometimes never appears at all, email isn't perfect, and sometimes it will bounce back to you. At times the machine that bounced the mail back to you will give you a hint as to what went wrong, but more often you're on your own.

The most common reason for a bounced message is a typo somewhere in the address. That's one reason that short email addresses are good—they're easier to type and thus less likely to be mistyped. The first thing to do when you get a bounce is to look through the header of the original message (the full headers are usually returned to you) and make sure you typed the address properly. Everyone makes this mistake on occasion.

A common error message in bounced email is "User unknown." This means that the email arrived at the proper machine, which searched through its list of users and decided that it didn't have a user with the name that you used. Again, this is most commonly due to a typo in the userid, although sometimes there are problems on the destination machine that have caused it to forget about a user, possibly temporarily. After checking the address, try resending the message, especially if you've gotten mail through to that address before.

Along with unknown users, you may see error messages complaining about "Host unknown." This is a more serious error that's more difficult to work around if it's not simply a typo. The basic problem is that for some reason one machine, perhaps your host machine or perhaps one further along in the path to the destination, was unable to contact the destination machine. There's not much you can do in this case other than try again a little later, in the hope that the machine you're mailing to has come back up.

One thing to watch out for is that sometimes the header provides an incorrect address to your email program, but the person includes the proper one in his signature. If you have trouble replying to a message, and the address you're replying to is different from an address listed in the signature, try using the signature address instead.

Note

I find this trick especially useful if the address in the signature is simple, whereas the address from the header is long and convoluted. Simple addresses seem to be more reliable, on the whole, although that's not a hard and fast rule.

Part II
7
Email Basics

Using Email Programs

All email programs share some basic features that you need in order to quickly and efficiently read, reply, and store your email. However, after these basic commonalities, the differences between programs mount quickly, so I concentrate on those differences when I talk about each program later in the book. For the moment, though, let's look at what an email program must do at a minimum.

Reading Email

An email program should enable you to display and scroll through an email message easily. Because you're using the Macintosh, you should be able to do all the standard

Macintosh things to text in a window, such as copying and pasting into a different program, resizing the window to display more text, selecting all the text with a single command, and that sort of thing.

Although you usually can choose the font and size in which you view messages in Macintosh email programs, I recommend that you stick to a monospaced font such as Monaco or Courier. People on the Internet must format tables and graphics with spaces, and monospaced fonts such as Monaco and Courier display these tables and graphics properly. Proportionally spaced fonts such as Times and Helvetica don't work as well because the characters in these fonts can have different widths, with a lowercase *i* being thinner than an uppercase *W*. Few of these features are generally available with a Unix mail program; that's one of the major advantages of using the Mac.

Note

Some email or communications programs come with special monospaced fonts that are supposedly easier to read than the standard Monaco and Courier. If you see fonts like TTYFont, VT100, or Mishawaka, they're designed for displaying Internet email on the screen. I personally prefer Monaco 9 point.

Navigating and Managing Email

Another important feature of an email program is the manner in which it enables you to move between messages, save messages in different mailboxes, and delete unwanted messages after you've read them. Most email programs display a list of the messages in an In Box area, and some indicate which messages you have already read, replied to, or saved to disk. Opening a message usually opens a new window to display the message, and sometimes closing the window (with or without deleting the current message) opens the next unread message—a nice feature for those who receive a great deal of mail. Being able to sort the list of messages is useful, and you should be able to select multiple messages at once to move them to another mailbox or delete them.

Note

Some of these navigation features may not exist in the email program you use, but with a macro program like CE Software's QuickKeys, you may be able to simulate them.

Speaking of multiple mailboxes, all email programs should support them, though unfortunately not all do. Most people want to save some of the messages they receive, so a program should allow you to create your own mailboxes for filing away messages on different topics. Of course, if you can create a new mailbox, the email program should enable you to do everything you can do in your In Box to the messages stored in a personal mailbox.

Perhaps the worst, and most surprising, offender in the multiple mailbox world is Apple's PowerTalk (released in late 1993), which puts a single mailbox on your Mac's desktop, but loses mail-related information like sender, subject, and date if you copy the messages out of the mailbox to a folder.

While you're managing your email, you will undoubtedly want to delete many of the messages you receive. This area may seem straightforward, but the better email programs follow Apple's lead with the Finder and enable you to trash a file without actually immediately deleting it. The easier it is to delete a message (and it should be very easy since you're likely to eventually want to delete most of the mail you receive), the more likely you will eventually delete something accidentally. If the email program deletes immediately, your message is toast. The other advantage of the two-stage (where a message is put in a trash can before being deleted later) or a delayed delete (where a message is marked as being deleted but isn't actually deleted until you close the mailbox) is that you then don't have to put up with an annoying confirmation dialog box every time you delete a message.

Some of your messages may, in fact, contain programs or other files that you want to save to a normal Macintosh file. A few email programs automatically detect attachments encoded in certain formats (more about file formats later) and decode such messages on their own. But one way or another, you need a simple way to save the message you're looking at without copying and pasting the entire thing into a word processor.

Replying to Email

Much of the mail you receive requires a reply of some sort, so an email program should make replying extremely easy, either with a command key shortcut or a single click on an icon. An email program also should facilitate *quoting* the original message, or prefixing each line with one or two special characters, usually a greater-than symbol and a space. Using quoting, you can easily include some of the message you're replying to so that the recipient has some context to know what you're talking about. A nice feature is the ability to select just a certain part of the original message and have the email program quote only that selected text in the reply and ignore the rest of the original message (see figure 7.2).

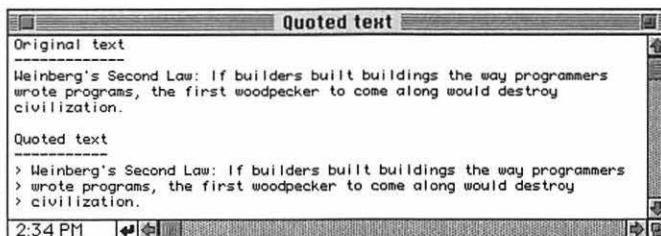


Figure 7.2 Original text and quoted text.

Note

Rick Holzgrafe has written a clever little \$10 shareware utility called SignatureQuote that enables you to quote messages properly even if your email program doesn't support it. SignatureQuote supports two quoting styles, two signatures that you can paste in, and can join and split text. It's a must-have if you're locked into a lousy email program. It's in:

<ftp://ftp.tidbits.com/pub/tidbits/tisk/util/>

Because an email message originally may have been sent to several people, an email program should give you the option of replying only to the sender or to all the people to whom the message was originally sent. At the same time, it ideally should make sure that you see the salient lines in the header. I've spawned a couple of embarrassing scenes by forwarding a message to a friend, and when my friend replied to me, his email program saw that mine had included the original message's address in the Reply-To line in the header. So his reply, instead of going just to me, went back to the sender, which was a mailing list that went to thousands of people. Oops! Luckily, I didn't say anything embarrassing and neither did he, so we were safe, but that's a good example of how two computer professionals who know better could have been thoroughly embarrassed in public. Think of this situation as standing up in a crowded restaurant and shouting loudly that your underwear has holes. You get the idea.

More powerful email programs provide features that can automatically mark or reply to email based on the contents of the header or the body of the incoming message. They often generalize these features so that you can essentially run a *mailserver*, which sends out requested information automatically via email. You also can use this sort of feature to run a simple mailing list, which takes a message to a certain local address and forwards it automatically to a list of subscribers.

Note

There are several programs that enable you run your own mailing list from a Macintosh and do things like auto-replies, although they currently require a direct connection to the Internet. Look for MailShare and AutoShare in:

<ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/mail/>

New Email

When replying to email and when creating new mail, an email program should provide all the features that you're used to when you're writing in a Macintosh word processor. In my opinion (which is by no means universal), the standard editors in the character-based Unix world stink (yeah, I know those are fighting words), and I spend so much time

writing and editing my email that I couldn't possibly put up with anything other than a decent Macintosh editor. However, because every email program implements text entry and editing on its own, none of them compare to a full-fledged word processor, and a few barely even compete with the free TeachText or its fancier successor, SimpleText. My dream, which isn't all that far off, is to be able to use Nisus Writer, the word processor with which I'm writing this book, for all of my email. With the growing acceptance of Apple events, this change will happen. I just hope it happens sooner rather than later.

I may be odd in this respect, but I think that any email program should make it easy to save a copy of everything you write, preferably automatically. I send more email than most people, often as many as 1,000 messages per month, but I like to be able to go back on occasion and see what I said, forward a message to someone who lost it, or just browse through the thoughts that appeared in my writings at that time. Why bother to keep a diary if you're writing about most of what happens in your life in email to friends?

Finally, whenever you create email, your email program should enable you to send the mail to a *nickname* or *alias*, which is merely another, easier-to-remember form of an email address. So instead of typing **ferdinand-the-bull@cork.tree.com** every time you want to send that person email, you can type the shorter **Bull**. Be careful with nicknames because it's easy to create more than you can easily remember, at which point they don't particularly help any more. Defining nicknames for everyone you might ever send email to is a waste of time; settle for defining a nickname only after you decide that you are likely to send that person email frequently.

Note

You want to be slightly careful with nicknames, because occasionally the recipient sees the nickname as well as the address. A friend once created a nickname DA BOSS for our supervisor, who thought it was funny when she saw it. I could think of some less humorous situations.

Finding People

Now you know how an email program should work and how to read email addresses when you see them littering up this book and the nets in general. But how do you find people to write to? Finding people to write to depends on what you're looking for. Hmm? What does he mean by that?

Assume that there are two types of people—those you already know and those you haven't yet met. The latter group makes up most of the world, and in some respects, they are the easiest to find and talk to because you don't really care who specifically you end up talking to. After all, you don't know any more about one stranger over another, so whom you talk to makes no difference.

Friends

When I first started using the networks way back when, few of my friends had accounts, and of them, only my best friend from high school ever managed to send me email more than once. I think I got a total of three messages from him. I tried to convince them, but I just couldn't get my friends to use email. Finally, I decided they all truly hated me (a logical conclusion for a 17-year-old college freshman) and gave up on them. My ego has recovered some since then, because I've found that convincing people to start using email just to talk to me is almost impossible. This argument worked with my parents, after a while, especially after my sister also started using email heavily at Cornell. But otherwise? I can't think of a single person whom I've convinced to use email for my sake, although we recently managed to set up my grandmother with an old Macintosh Classic and an email account. The moral of the story is that you should assume that you can talk only to people who already use email.

Okay, so once past that reality check, how do you find the address of someone who you know uses email? The simplest and most effective method seldom occurs to many net denizens—use the telephone and ask them. This method, low-tech though it may be, has the advantage of being quick, accurate, and easy. Of course, it does ruin the surprise value of that first email note. Such is life. You do need to know your friend's telephone number, or failing that, her address so you can call the all-knowing information computers at the phone company. If you don't even know where your friend lives, she may be trying to hide from you anyway after that ugly incident awhile back.

"Aha!" you say, "If the all-knowing phone company computers can give me my friend's telephone number, aren't there all-knowing computers on the Internet that can give me my friend's email address?" Nice try, and good question, but the answer is, unfortunately, maybe. Some computers know what users they support, and you can find some information via services called *Finger*, *Whois*, *X.500*, *Ph*, *Knowbot*, and various others, but that information doesn't help unless you already know what machine to search. Several attempts have been made at linking various directory services on different machines, but I've never found them to be the slightest bit useful. The problem is twofold. First, hooking a local directory of users to an Internet-wide directory requires some effort and certain standards, and inertia being what it is, that effort isn't always made, and the standards don't exist. Second, many organizations shield their users from the outside world for reasons of security and privacy. These shields also make it difficult to determine how many people actually use the Internet because one domain may have two users, like **tidbits.com**, or many thousands of users, like **microsoft.com**.

Note

Frankly, because I find these services so completely useless, I'm not going to bother to discuss them further. That said, if you crave some frustration, go to the University of Minnesota's Gopher server at **gopher.tc.umn.edu** (the Home Gopher server by default, if you use TurboGopher), select Phone Books, and then check out the various different options available

*for searching for Internet email addresses. If you know the organization in question, and they have a Phone Book server, that's the best start—otherwise you're on your own. The main thing you miss via the Gopher route is the Knowbot Information Service. To access it, telnet to **info.cnri.reston.va.us 185** and type **help** to see the possible commands.*

Acquaintances

As I said earlier in this chapter, finding new friends is easy on the Internet. You don't know people beforehand, so communicating with them in a discussion list via email or news requires nothing in terms of opening lines or trivial small talk about the fallibility of weather forecasters. If you have something to contribute to a discussion, or perhaps if you merely want to make a private comment to one of the people in the discussion, meeting him is as easy as replying to his message. Whether that first contact grows beyond a one-time message depends on many variables, but with so many people, finding correspondents on the net doesn't take long.

As much as meeting people may be easy, finding them again after some time often proves more difficult. You may not remember where a person lives, if you ever knew, and if it's in the United States at all, you probably don't know his telephone number; and frankly, you may not even remember how to spell his name. And yet, all too often I've had long, involved conversations that eventually trail off after several weeks or months, and then I don't hear from that person again. If I haven't saved a message (which contains the all-important email address in the header) or recorded his email address somewhere, I have to hope that my friend has better organizational systems than I do.

I suggest that you figure out some way to keep track of your correspondents' email addresses. Nickname features work well although they may prove unwieldy as a storage mechanism later on. If that's true, I recommend using a standard database or address book program that can handle extra fields for email addresses. This advice may sound obvious, but I can't tell you how many times I've lost an address that I wanted several months later. These days I keep a copy of every piece of email I send, in part so I can search that file, large though it may be, for email addresses that have escaped my short-term memory.

Mailing Lists

There's no accounting for taste, and similarly, there's no accounting for different interests. I may be interested in electronic publishing, tropical fish, and competitive distance running, whereas the next person might favor *The Simpsons*, aviation, and Irish culture. As a result, discussion groups have sprung up around almost every imaginable topic, and if your area of interest isn't represented, it's not too difficult to start your own group. These

groups take two forms: mailing lists and Usenet newsgroups. I talk more about Usenet in the next chapter; for now I'll concentrate on mailing lists.

The beauty of mailing lists is that they cover specific topics and they come straight to you, without any extra work on your part. If you find yourself interested in a topic, you can subscribe to the appropriate mailing list, and all the traffic comes directly to your electronic mailbox. This system makes participating in many mailing lists easy, even if you have only email access to the Internet; Usenet access may require more money and effort. Luckily for those of you who cannot get Usenet access, many mailing lists and newsgroups mirror each other.

Mailing lists have several other advantages over Usenet news. Email is ubiquitous on the Internet, whereas access to news is less common (although certainly widespread). Because of the way Usenet news propagates throughout the nets, mailing lists often arrive faster than any given posting in a newsgroup. Because mailing lists arrive in your electronic mailbox, they may seem less intimidating than large newsgroups with many participants. And frankly, many of us who lead busy lives find mailing lists easier to keep up with because we don't have to run another program to read the list, whereas reading news always requires leaving that ubiquitous email program and then running a newsreader.

There are a number of programs that operate mailing lists, the most well-known of which are LISTSERV, ListProcessor (also known as listproc), and Majordomo. They all support similar commands; I'll get into those in a moment. LISTSERV is a commercial program from Eric Thomas of LSoft. It currently requires an IBM mainframe running VM/CMS (although versions for VMS, Unix, and Windows NT are in the works). The Unix-based ListProcessor comes from Anastasios Kotsikonas and is now owned by CREN (remember them from chapter 4?). The Unix-based Majordomo is free as far as I can tell.

There also are many mailing lists that are run through hacks to the Unix mailer software—these generally require some sort of human intervention for subscribing and signing off, although sometimes they use non-standard commands that do the same thing. Have I mentioned yet that I dislike programs that don't work in standard ways? They make life even more confusing than it already is.

Note

You may wonder why LISTSERV doesn't have an E at the end and why it is spelled with all capitals. LISTSERV software has existed for some time on IBM mainframes that run the VM/CMS operating system. This operating system limits userids to eight characters (hence the missing E), and because the operating system itself was originally not case sensitive, all commands and program names have traditionally been typed in uppercase only. The name also may have had something to do with early computer terminals not supporting lowercase, but I can't prove that theory. Just believe me—by convention, LISTSERVs are always addressed in the uppercase, although it doesn't matter anymore.

Along with the different mailing list manager programs, you may have to deal with two other variables related to mailing lists—moderation and digests. Each of these possibilities slightly changes how you interact with the list, so let's look at each in turn and then go over the basics of using the list manager software as a subscriber.

Moderated vs. Unmoderated

I suspect many mailing lists started out *unmoderated*, which means that anyone was able to send a message on any topic (whether or not it was appropriate to the group) to the list. The list software then distributed that message to the entire list. You see the problem already—no one wants to read a bunch of messages that have nothing to do with the topic or discussion at hand. Similarly, if a discussion is spinning out of control and turning into a flame war, it's just a waste of time for many people.

Thus was born the concept of the moderated mailing list. To stem inappropriate postings, a moderator reads all the postings before they go out to the group at large and decides which are appropriate. Moderated groups tend to have less traffic, and the messages that go through are guaranteed by the moderator to have some worth. This system is good.

Note

The Info-Mac Digest is a prominent example of a moderated group in the Macintosh world. Although they're usually fairly lenient, moderators Bill Lipa, Gordon Watts, Igor Livshits, and Liam Breck do an excellent job, and their efforts are much appreciated by all. I help out somewhat with Info-Mac, although with the files stored on the Info-Mac Archive, not with the Info-Mac Digest itself.

On the downside, moderated groups occasionally run into sticky issues of censorship because the moderator may not always represent the views of the majority of the readers. Moderator positions are volunteer only; I've never heard of a mailing list that elected a moderator, although it's certainly possible, particularly among lists that carry traffic associated with a professional organization.

There's a slightly different form of moderation practiced on some other lists that you may be interested in, such as the Apple Internet Users and Apple Internet Providers lists. Run by Chuq von Rospach of Apple, these lists employ a "List Mom" form of moderation, which means that although all postings appear whether or not Chuq approves, he reserves the right to tell people to take off-topic discussions elsewhere. Since I was about to start similar lists when Chuq created the Apple Internet Users and Providers lists, I volunteered to be an Assistant List Mom on Chuq's lists, which means I help steer the discussions so that they stay on track. Needless to say, I recommend both lists highly for discussions about the Mac and the Internet.

Note

To subscribe to the lists, send email to **listproc@abs.apple.com** and in the body of the message (the Subject line can be blank or nonsense), put **subscribe apple-internet-users Your Full Name** or **subscribe apple-internet-providers Your Full Name**.

Once your subscription has been processed, you will receive an informational file from the list, and then you can start posting questions and helping out with answers that you know. You cannot post to the list until you have subscribed.

I see no reason to choose to read or not read a mailing list based on its moderation until you've spent a while seeing what goes on in the group. I subscribe to various lists, some moderated, some not, and on the whole, both have their place. Keep in mind, though, that if you post to a moderated list, the moderator may reject your posting. Don't feel bad, but do ask why so that your future submissions stand a better chance of reaching the rest of the group. On the other hand, when posting to an unmoderated group, try to stick to appropriate topics because people hate hearing about how you like your new car in a list devoted to potbellied pigs. Too many misdirected postings to a list may agitate list members to the point of asking for a moderator to limit the discussion.

Individual Messages vs. Digests

When the number of messages in a mailing list increases to a certain level, many lists consider creating a digest version of the list. A digest is simply a single message that contains all the individual messages concatenated in a specific way. Why bother with a digest? Depending on how your email program works, you might find it awkward to receive and read as many as 30 messages a day, especially if your email service, such as AppleLink, charges you a per-message fee to receive email. Just think how many messages you may have waiting after a week of vacation. If the messages are sent in digest form, a mailing list becomes easier to handle for some people because you get one big message instead of lots of little messages.

Unfortunately, digests have problems too. Some email gateways to commercial services (again AppleLink) limit the size of incoming email messages. Thus, digest mailing lists like the Info-Mac Digest, one of the most popular Macintosh mailing lists, can range in size from 30K to over 100K, so very few issues of the digest sneak through the gateways with size limitations. In addition, you may find it easier to read (or skip through) small individual messages, whereas scrolling through a 100K file can take quite a bit of time and can be extremely awkward with some email programs. To add to the complication, certain email programs can break up a digest into its individual messages for easier viewing. I'm talking the email equivalent of digestive enzymes here.

You must decide for yourself whether a digest is easier or harder to work with, but only with some groups do you have any choice. The LISTSERV and ListProcessor software

sometimes provide an option that you can set to switch your subscription to a mailing list from individual messages to a single, usually daily, digest. I don't believe you can toggle this option for Majordomo-based lists, but Majordomo list administrators can set up a separate list that sends out a digest—you would simply subscribe to the separate (digest) list instead. These separate digest lists in Majordomo generally have “-digest” appended to the listname.

Mailing List Managers

Mailing list managers sport many sophisticated features for managing large mailing lists, and these features have made the programs popular among the people who start and run mailing lists (you didn't think lists just worked on their own, did you?). For instance, you can easily and automatically subscribe to and sign off from mailing lists run with a mailing list manager without bothering a human (in most cases). This significantly reduces the amount of work that the list administrator has to do. These programs generally also have provisions for tracking the subscribers to a list, and for those who want to remain unknown, concealing certain subscriptions.

Mailing list managers can prevent unauthorized people from sending messages to the list. The *TidBITS* list works this way in theory because only I can send a message, in this case an issue of *TidBITS*, to the list. I say “in theory” because in practice the safeguards have broken down twice, resulting in confusing messages going to the entire list. The *LISTSERV* that runs the *TidBITS* list also knows to route all replies to postings on the list to me directly, which is normally good, but when these two accidental postings got through the safeguards, I received hundreds of messages from confused readers who didn't know why they had gotten this message. It was a major hassle.

The *LISTSERV* software knows about other *LISTSERVs* running on other machines around the world and uses this knowledge to limit network traffic. For instance, I send a single message from my machine to a mainframe at Rice University in Texas that runs the *LISTSERV* handling the *TidBITS* list. Once the message arrives at Rice, the *LISTSERV* software checks to make sure it came from me and then sends it out to the many thousands of readers on the *TidBITS* subscription list.

The *LISTSERV* is smart, however. It doesn't blindly send out thousands of messages, one per user, because that would waste network bandwidth, especially on expensive trans-oceanic links. Instead, the *LISTSERV* determines how to enlist the help of certain other *LISTSERVs* running around the world. If it knows of a *LISTSERV* site in Australia, for instance, it sends a single copy of the message to Australia along with the list of Australian readers to distribute to. If 100 people in Australia subscribe to the *TidBITS* list, only one message crosses the Pacific instead of 100 identical copies of the same message. That's elegant.

I gather that *ListProcessor* and *Majordomo*, Unix mail-based mailing list managers, don't have as many features as the *LISTSERV* software, but that's more from an administrator's

point of view. Users generally shouldn't care which they use. Don't worry about it one way or another; you have no choice when picking mailing lists to subscribe to. And despite the added features of the LISTSERV software, the thing that makes the administrative details of a list easy to deal with is the administrator, not the software.

Note

ListProcessor used to be called Unix-Listserv and was distributed freely (version 6.0c is still free, even now that CREN has bought it and will be selling future versions), but after some unpleasantness regarding the term "Listserv," Anastasios Kotsikonas decided to rename it to avoid confusion. So, if you see references to Unix-Listserv, they're talking about ListProcessor.

Using LISTSERVs

Most people find dealing with LISTSERVs quite easy; however, you should watch out for a few common pitfalls while working with LISTSERV-based mailing lists. Many of the following commands and pitfalls apply to lists run by the other mailing list managers as well, so it's worth reading through the LISTSERV section even if, for instance, you're dealing with a ListProcessor list.

Every LISTSERV list has two email addresses associated with it: the address for the LISTSERV itself, and the address for the mailing list. Why the dichotomy? Well, the LISTSERV address handles all the commands, things such as subscriptions and requests for lists of subscribers and the like. The mailing list address is where you send submissions to the list, assuming of course that it's that sort of list. Here, I use the *TidBITS* list as an example for my illustrations of the basic tasks you do with a LISTSERV-based mailing list. (The only difference between the *TidBITS* list and many others is that if you send mail to the mailing list address, it doesn't go to everyone else on the list because the *TidBITS* list is dedicated to distributing *TidBITS*, not to discussion, as are most lists. Any mail sent to the mailing list address comes to me, which is fine, because such messages are usually comments on articles.)

If you want to send the LISTSERV that handles the *TidBITS* list a command, such as your subscription request, send it to **listserv@ricevm1.rice.edu**. Notice that nowhere in the address is *TidBITS* mentioned, which is a hint that you have to specify *TidBITS* somewhere else. LISTSERVs ignore the Subject line entirely, so don't worry about filling it in at all. In the body of the message, though, you can put one or more commands, each on its own line.

To subscribe to the *TidBITS* list, you send the preceding address a message with the following command on one line in the body of the message: **SUBSCRIBE TIDBITS *your full name***, where you replace *your full name* with your real name, not your email address or some cute nickname. If the list administrator has to contact you about a problem, she

probably doesn't appreciate having to address you as "Dear Swedish Chef Fan Club Ork Ork Ork." To clarify the preceding command, to subscribe to any LISTSERV mailing list, you send the **SUBSCRIBE** command, a space, the name of the list you want to subscribe to, a space, and then your full name, which must be at least two words (see figure 7.3). I don't know how rock star types like Cher or Prince manage with LISTSERVs, although I did once get mail from someone who really only had one name. I advised him to use "No really, I only have one name" as his last name.

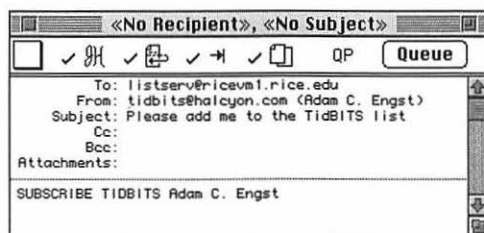


Figure 7.3 Subscribing to TidBITS.

Note

Prince has legally changed his name to a single character that looks like a little stick figure person. Such characters are called dingbats in the publishing world, so I vote that we simply refer to Prince as Dingbat, or perhaps Prince Dingbat to be unambiguous. Needless to say, there's no way to subscribe to a mailing list using a single non-alphabetic character as your name.

The LISTSERV always returns a welcome note after you have subscribed successfully. *Keep that note!* It lists various useful commands, such as how to sign off from the list, and it usually provides the address of the list administrator. You can contact the list administrator to handle any problems that the automated program chokes on.

You know what I just said about keeping the welcome message from the LISTSERV. Well, I really mean it! I can't tell you how irritating it is to have someone post the list asking how to sign off when it would have been trivial if they'd just kept that welcome message. Actually, I can tell you how irritating this is, because after the 10th or 100th person does, it's tremendously exasperating! Sorry, I'll quiet down now.

After you have subscribed to a list, you mainly want to read messages (which is easy) and post occasional messages to the rest of the people on the list. Once again, to post a message you send it to the mailing list address, which is always the name of the list at the same machine. If, for example, you want to send mail to the *TidBITS* list (which comes only to me), you send the email to **tidbits@ricevm1.rice.edu**. I realize I've almost beaten this particular horse to death, but I can't emphasize enough the difference between the LISTSERV address and the mailing list address. You send commands to the LISTSERV, and submissions to the mailing list address. Perhaps the most common problem I see on LISTSERV mailing lists is that people forget to send commands to the

LISTSERV address and instead fill up the mailing list with the electronic equivalent of junk mail that no one wants to see. And, of course, sending commands to the mailing list address isn't just annoying and flame-provoking, it's futile because the LISTSERV doesn't respond to them there.

After you've been on a list for some time, the LISTSERV may ask you to confirm your subscription. I set this option with the *TidBITS* list to clean the deadwood from the list every year. Students graduate, employees move on, bulletin boards close up, and those addresses don't always go away, so the LISTSERV wastes network resources sending to a nonexistent person. It's much like talking to a politician. After you have received *TidBITS* for a year from the LISTSERV, it sends you a message asking you to confirm that you still want to get the newsletter. If you don't respond within seven days, the LISTSERV removes you from the list, assuming that you don't want to continue receiving email from it. If you respond, you must respond with a command so that you send it to the LISTSERV address, not the mailing list address. The command is simply **CONFIRM TIDBITS** (or the name of whatever list you are asked to confirm).

A portion of the time this confirmation process fails. As I'm sure you noticed in the preceding paragraphs, nowhere do you provide your email address to the LISTSERV, which is supposed to determine it from the header of your message. This idea seems excellent at first because the header should, in theory, have your email address correct, and it doesn't suffer from typos or simple human mistakes that you make if you type it in by hand. However, depending on the routing that your mail takes and how you or your system administrators have your system set up, your address as it appears in the header may change from time to time. Those changes play havoc with the LISTSERV, which is a very literal program. Therefore, when you confirm a subscription, if that confirmation comes from an address the LISTSERV doesn't recognize, poof, it doesn't work. You probably still receive mail to the original address just fine because the address is usually merely a variant on the theme, so many people sit by helplessly as the LISTERV asks for confirmation, rejects it, and then calmly deletes them from the list.

Note

This situation is a perfect example of why computers should never be given direct control over human lives. If you don't properly match for some reason, you're just another file to be deleted.

There is a simple fix. Just resubscribe as soon as the LISTSERV sends either the confirmation rejection or the message saying that it has deleted you from the list. You may get duplicates of everything for a few days, but then the LISTSERV deletes your old address and continues to send to your new one.

If you blow it and misspell your name while subscribing, or perhaps decide to change your name for one reason or another, you can always change your name (only) with the

LISTSERV by sending another **SUBSCRIBE** command. The danger here, as discussed in the preceding paragraphs, is that if your address looks at all different from when you originally subscribed, the LISERV happily adds you to the list again, and you receive duplicates of everything. Now is a good time to ask the list administrator for help because the LISERV recognizes only your new address, so you can't delete your old address. Bit of a Catch-22 there.

This Catch-22 can apply to trying to sign off from a list normally, as well. Under standard circumstances, if you send the command **SIGNOFF TIDBITS** to the LISERV address, it removes you from the list. If your address in the header has changed, however, it doesn't recognize you as a current subscriber and thus doesn't let you sign off. Once again, if you need help beyond what the LISERV program can provide, don't hesitate to ask the list administrator, but ask nicely. These people don't get paid to take abuse, and in fact, they don't get paid to administrate a list at all. I'll tell you how to contact a list administrator in a moment.

The reason for this seemingly irritating address feature is that administrators realized early on that it would be way too much fun to sign someone else up for mailing lists if you really don't like him. You could, for example, sign him up for all the special offers in the back of *The National Enquirer*. Some friends of mine once had a war with that game, but one was declared the loser when he received bronzed baby shoes and a free subscription to a white supremacist newsletter, or some such nonsense. I'm sure it would be great fun to sign up Bill Gates for a really far-out mailing list, but it gets old after a while and is generally considered abuse of the networks.

Note

*Some LISERVs can send you files if you send them proper commands in a message. The LISERV at Rice, **listserv@ricevm1.rice.edu**, is one of these sites. In fact, it stores Macintosh files that also exist on the popular FTP site **sumex-aim.stanford.edu**. You can find site-specific information by sending a **HELP** command to any LISERV, and for the standard LISERV information, send **INFO REFCARD**.*

LISERVs support a number of other commands, of which only a few are generally useful. If you want to see a list of all the people who have subscribed to a LISERV list, you can use the **REVIEW** command, although many lists no longer return the full list of subscribers to protect against abuses.

The other utility of the **REVIEW** command is that it includes the address of the list administrator at the top, so it's a good way to find out who to ask for help. Using the **REVIEW** command is a good way to see what address the LISERV thinks you used to subscribe and then ask the administrator for help. For just the administrator address, you can change the command to **REVIEW SHORT**.

Note

Budding direct marketers should be aware that if you request a bunch of subscriber lists and use them for nefarious marketing purposes, the following will occur: (a) that feature will be immediately turned off, and (b) I will personally lead the flamethrower crews on a mission to turn you into a fine electronic ash. That sort of opportunism doesn't fly on the Internet.

To switch a LISTSERV subscription (you must already be subscribed) from individual messages to a digest format, send the LISTSERV address the command **SET *listname* DIGEST**. To switch back to individual messages, send it command **SET *listname* MAIL**.

Most of the other commands that LISTSERVs support aren't as interesting, or as much fun to write about, so I'll refrain and let you find them on your own.

Using ListProcessor

Working with a mailing list run by the ListProcessor program is remarkably like working with a mailing list run by the LISTSERV program. The similarity isn't coincidental—ListProcessor started out as a clone of LISTSERV, not in terms of the code, but in terms of the command structure. Thus, the few differences between the two are minimal, especially in the basic functions.

Just as LISTSERVs have a **listserv@domain.name** address, ListProcessors are generally referred to as **listproc@domain.name**, although a number of them may still use the **listserv@domain.name** address left over from when ListProcessor was called Unix-Listserv. And just as the mailing list itself has a different address from the LISTSERV address, something like **listname@domain.name**, so too do ListProcessor lists. In other words, the confusing dichotomy between the ListProcessor address and the list address exists, just as it does with LISTSERV lists. You send commands to the ListProcessor address (in the body of the message—the Subject line doesn't matter) and submissions to the mailing list address. I'm really beginning to feel sorry for this poor horse, since I keep beating it, but I can't tell you how many people fail to understand this basic distinction, and in the process irritate thousands of other people on numerous lists.

To subscribe to a ListProcessor-based mailing list, send **subscribe *listname* your full name** to the ListProcessor address. Just as with the LISTSERV mailing lists, replace *listname* with the name of the list you wish to subscribe to and use your real name in place of *your full name*. ListProcessor figures out your email address from the header of the message.

You leave a ListProcessor-based mailing list by sending the command **unsubscribe *listname*** to the ListProcessor address. The command **signoff *listname*** does exactly the same thing. Just like the LISTSERV lists, if your address has changed, the automated process very well may not work, at which point you must talk to the list administrator.

The command to switch a ListProcessor subscription from individual messages to digest format differs slightly from LISTSERV—send the command **set *listname* mail digest** to

the ListProcessor address. Frankly, I can't see from the instructions how to switch back to individual messages.

If all else fails, try sending the ListProcessor the **help** command for a simple reference card that explains the options.

Using Majordomo

This is getting kind of boring, but Majordomo works pretty much like the other two mailing list managers. There are two addresses—the Majordomo address to send your commands to (often **majordomo@domain.name**), and the mailing list address to send submissions to (**listname@domain.name**). You also (if they're running a recent version of Majordomo) may be able to send commands to **listname-request@domain.name**.

To subscribe to a Majordomo-based list, send email to the Majordomo address with the command **subscribe listname**. Majordomo differs slightly from the other two mailing list managers in that you don't have to specify your full name, and if you like, you can append an email address to the subscription command. This enables you to subscribe someone else to a mailing list, which can be handy—just don't abuse it. The same structure applies to removing yourself or someone else from a list—send **unsubscribe listname** to the Majordomo address (**signoff listname** works as well).

An easier method of subscribing and unsubscribing to Majordomo-based lists is to send email to **listname-request@domain.name** with either the **subscribe** command or the **unsubscribe** command in the body of the message. Because you've made it clear which list you want to subscribe to with the address, there's no need to include it in the subscription command.

Finally, you can send Majordomo a **help** command to see what other options are available. I always recommend that you do this, just so you know how and so that you see what's possible.

Neither Rain, Nor Snow...

Because you're likely to use email heavily, I hope you've gotten a sense for how it works, the sorts of things you shouldn't do with it, what an email program should do for you, and what it makes possible in terms of mailing lists. There are thousands of mailing lists available on the Internet, and you can find some wonderful discussions. If you'd like to search for a good one, check out this URL via a Web browser:

<http://alpha.acast.nova.edu:80/listserv.html>

But, enough about mailing lists—let's move on to the next sort of discussion lists, the Usenet newsgroups.

Chapter 8



Usenet News Basics

I've talked generally about the thousands of Usenet newsgroups that hold fast-moving discussions on every imaginable topic. My host machine, for example, carried over 5,000 of them at last count, and that's nowhere near the entire list, which is closing in on 10,000. Hundreds of thousands of people read Usenet every day. It's certainly one of the most interesting, although strange, parts of the Internet.

Note

Prompted by a problem posed by Nicholas Negroponte, head of the MIT Media Lab, Eric Jorgensen of MIT did a survey in early 1994 to determine the average age and gender of Usenet readers. Jorgensen received 4,566 responses to his survey. He figured out that the average age of the Usenet reader is 30.7 years old (with a standard deviation of 9.4). Eighty-six and a half percent of the replies came from men, 13.5% came from women, and 0.1% came from, well, not men or women. Although most newsgroups he surveyed were heavily male-dominated, misc.kids (71% female), rec.arts.tv.soaps (91% female), and rec.food.sourdough (50% female) were notable exceptions. You may be able to find more information about the survey and the full results in:

<http://www.mit.edu:8001/people/nebosite/home-page.html>

How is Usenet different from the mailing lists we've just looked at in the last chapter? I see two primary differences, neither of which has to do with the information that flows through them.

First, although mailing lists may be faster to propagate because they go directly to the subscriber, they can be extremely inefficient. If only one person on a machine reads a mailing list, one copy comes in. If, however, 100 people on that machine all read the same mailing list, then 100 identical copies of each posting must come in, eating up disk space and slowing down other tasks. This is bad. In contrast, only one copy of every Usenet message goes to each machine, and any number of people on that machine can read it. So, assuming that both contained an identical posting (which in reality occurs only occasionally), you could greatly reduce your machine's storage load by reading the Usenet news group instead of the mailing list.

Second, many people like mailing lists because they always read their mail but may not always run a separate newsreading program. This situation actually works in favor of news as well. Most email programs are designed for a relatively small number of messages, each completely different and unrelated. In contrast, most newsreaders concern themselves with large numbers of messages, many of which are related, or in a *thread*. So, if you read the news and come across an interesting posting, reading the next posting in that thread is easy (or should be), whether or not the posting is the next one in the list. Following threads in an email program is generally difficult or impossible.

Given those advantages, how does Usenet work, what do the messages on it look like, and how do you generally interact with it?

Usenet Plumbing

For the most part, knowing how Usenet actually works isn't even slightly important to daily life. However, the basic principles may help you to better cope with some of its quirks and limitations.

The entire concept of Usenet is based on one machine transferring postings to another. Scale that up so that any one machine carrying Usenet messages talks to at least one other machine carrying Usenet messages, and you start to see how this simple idea can become an immense and powerful reality. We're talking about thousands of machines and millions of people and hundreds of megabytes of data per day.

If you post a message in a Usenet group, your machine passes the message on to all the machines it talks to, both *upstream* and *downstream*. Upstream loosely refers to the machines that your machine generally gets all of its news from. Downstream loosely refers to the machines that get all of their news from your machine. In either case, those machines continue to propagate your message throughout the network, with the Usenet software that controls the system making sure that your message isn't duplicated *ad infinitum* (Latin for "a hell of a lot of times, which irritates everyone").

The actual process by which your message travels is equally simple, at least in UUCP. The Usenet software creates a batch of messages and compresses the batch to reduce transmission time. When the next machine receives the batch, it unbatches the messages and places the files in directories known by the news-reading software. One testament to the simplicity of this scheme is that not all implementations have to use this technique. (In fact, NNTP, another common method of transferring news, sends only the text of articles that a specific reader requests while reading news.) InterCon's UUCP/Connect on the Macintosh, for example, creates a single file for each newsgroup and appends new messages to that single file (which is much more efficient given the way the Macintosh file system works). However, most Unix machines store the messages as individual files within specific directories, and those directories are directly related to the names of the newsgroups.

Newsgroup Innards

Just as email addresses make sense after you know all the parts, so do the Usenet newsgroup names. Although they resemble email addresses, the basic principles are a bit different.

Note

*Although, like email addresses, Usenet newsgroups use periods to separate different parts of the name, people tend not to use them in conversation. If, for example, you were to tell a friend about an interesting discussion on **comp.sys.mac.misc**, you'd say, "Check it out on comp sys mac misc." Part of the problem may be the linguistic clumsiness of saying all those "dots," but I suspect more of the reason is that precision isn't nearly as necessary. Unlike email addresses, you seldom type out newsgroup names. It also may have to do with the fact that newsgroup names are all unique and easily parsed.*

The premise of the Usenet newsgroup naming scheme is that of a hierarchy. The naming scheme makes figuring out how to name new groups easy. More important, it maps over to a hierarchical directory (or folder) structure. On the Unix machines that hold the newsgroups, therefore, you find a directory called **news**. Inside that directory are other directories corresponding to the top-level parts of the hierarchy—**alt**, **comp**, **misc**, **news**, **rec**, **sci**—and so on. These directories are abbreviations for alternate, computers, miscellaneous, news, recreation, and science, respectively.

Note

*I could attempt to create a table listing all the top level hierarchies, but it's a pointless task. There are many local hierarchies that I have no way of finding (just as many other machines probably don't carry the **halcyon** or **seattle** hierarchies that I can see), and I couldn't begin to guess which hierarchies your machine might carry.*

Let's dissect the name of **comp.sys.mac.misc**, a popular newsgroup. If we first look into the **comp** directory, we see more directories corresponding to **lang**, **sys**, and so on. Under **sys** we find many directories, one for each computer system. There are **atari**, **amiga**, **ibm**, **mac**, and gobs of systems that you may never have even heard of. (I certainly haven't heard of all of them.) After we go into the **mac** directory, we find the lowest level directories that correspond to the individual topics about the Mac. These include **advocacy**, **apps**, **databases**, **games**, **hardware**, **misc**, **portables**, **system**, and others. Once inside those directories (feel like you're in a Russian doll again?), you find the files that hold the text of the messages (see figure 8.1).

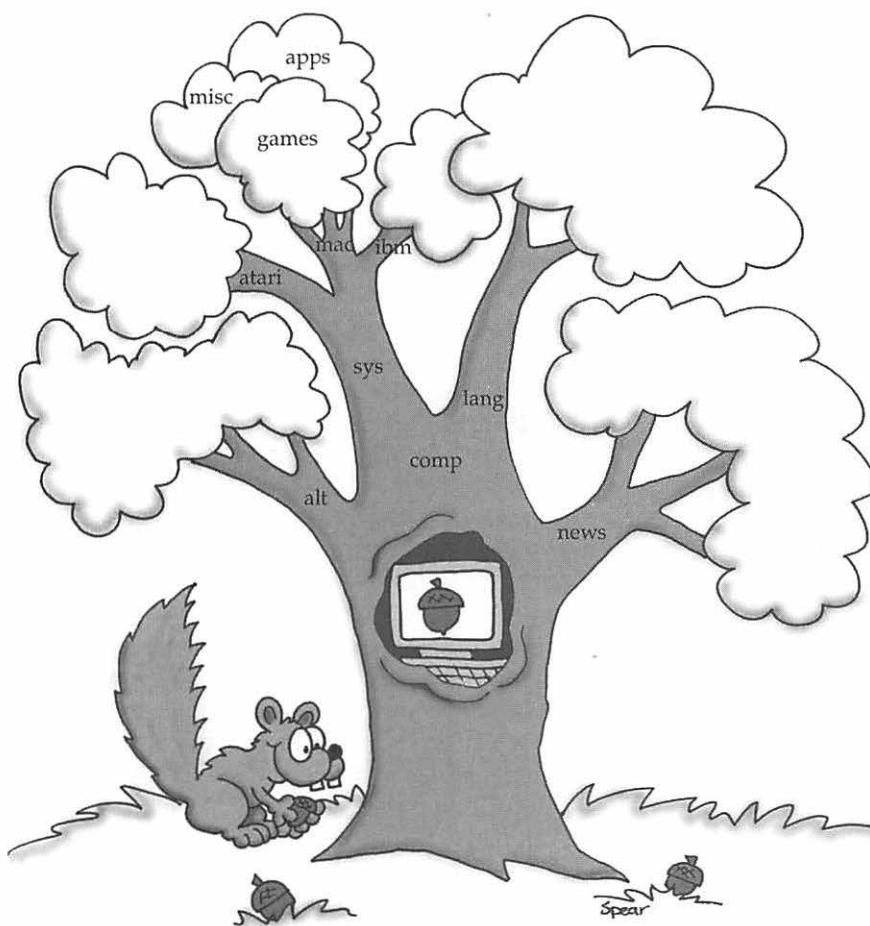


Figure 8.1 *An abbreviated Usenet hierarchy tree.*

This system may seem a tad clumsy, but remember, as a user you never have to traverse that entire directory structure. It exists to categorize and classify newsgroups, and to provide a storage system that maps onto a Unix directory structure.

Historically, this structure was used so that as little C code programming as possible had to be done to store and retrieve Usenet news. A design goal for B-news and C-news, the earliest Unix news servers, was to make as much of it as possible run as Unix shell scripts, precluding any fancy binary database backends. Why was that? Laziness is the mother of some invention.

Message Construction

On the surface, a Usenet posting looks much like an email message (see figure 8.2).

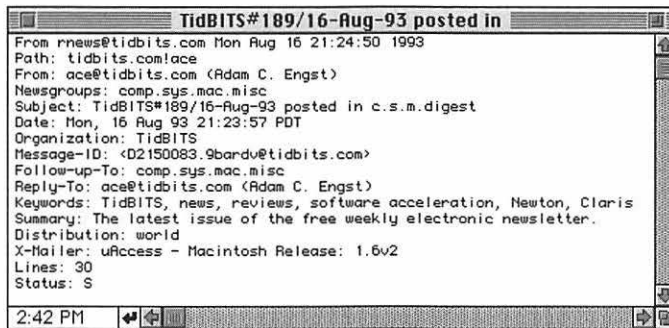


Figure 8.2 A Usenet header.

The posting's header holds a From line, a Subject line, and a fair amount of other stuff; next comes the body of the message and a signature.

The header has a few new lines that you may find interesting.

```
Newsgroups: comp.sys.mac.misc
```

First comes the Newsgroups line. It lists, separated by commas, all the newsgroups to which the message is posted. You can post a message to more than one group at a time by putting more than one group in the Newsgroups line. At that point, an article is *cross-posted*. If you must post an article in several groups (which is generally frowned on as a waste of bandwidth), make sure to post via the Newsgroups line and not through individual messages. Individual messages take up more space, because a machine stores only one copy of a cross-posted article along with pointers to it from different groups.

Follow-up-To: comp.sys.mac.misc

The next Usenet-specific line is the Follow-up-To line, which usually contains the name of the newsgroup in which the article appears. Sometimes, however, you want to post an article in one group, have a discussion, then move back to another group. In this case you put the second group in the Follow-up-To line, because whenever anyone posts a follow-up to your article, the news software makes sure that it ends up in the proper group.

Reply-To: ace@tidbits.com (Adam C. Engst)

When people reply via email to a newsgroup posting, their newsreaders use the address in the Reply-To line. A Reply-To line makes it easier for people to respond directly rather than cluttering the group with personal messages or those that aren't relevant to the group (especially any flames).

Keywords: TidBITS, news, reviews, software acceleration, Newton, Claris

Summary: The latest issue of the free weekly electronic newsletter.

Sometimes you see a Keywords and/or Summary line as well. Although they aren't universal or enforced, it's often a good idea to fill in these lines for your article before you post it. That way, people who have set up their newsreaders properly can more effectively filter articles based on keywords. In addition, some newsreaders show only the header and first few lines of an article, and then let the reader decide whether she wants to read the whole thing. A few well-chosen keywords or a concise summary can help make that decision easier.

Distribution: world

Many articles are only relevant in specific geographic areas. You have two ways to handle this situation. First, if you're selling a car in, say, Seattle, you should post to a specific group that goes only to people in Seattle (more or less, anyway), such as **seattle.forsale**. Many of these site-specific groups exist, even down to the machine. There's a group, **halcyon.slip**, for example, for discussion about issues affecting SLIP and PPP users of the **halcyon.com** machine.

The other way to handle this situation is to use the Distribution line. This enables you to limit the area to which your message is distributed, even if the group encompasses all of Usenet. So, if you want to post a notice about a Seattle British Car Show in **rec.autos**, you should put **seattle**, or possibly **pnw** (for Pacific Northwest), in the Distribution line.

Subject: TidBITS#189/16-Aug-93 posted in c.s.m.digest

And of course we have the ubiquitous Subject line. Much as it is courteous for you to provide a descriptive Subject line in an email message, it's imperative in a Usenet posting. Most newsreaders these days show the user a list of the messages and their subjects. If you don't provide a good Subject line, far fewer people even notice your message. For example, each week I post an announcement of each issue of *TidBITS* in the **comp.sys.mac.misc** group. Instead of a general Subject line such as "TidBITS posted," I enter "TidBITS#189/16-Aug-93 posted in c.s.m.digest." This tells the reader in precise terms what I posted and where he can find it.

The Newsgroup Stork

Now that you know something about how messages travel from machine to machine and how the naming system works, you may wonder where new newsgroups come from. Whenever I've talked about the range of Usenet groups, I've said something to the effect of "and if there isn't one that matches your interests, you can create one." That is true, but the process is not trivial.

Note

*Although I summarize the process in this section, if you want to see all the gory details, check out two periodic postings in **news.announce.newusers**. Both "How to Create a New Usenet Newsgroup" and "The Usenet Newsgroup Creation Companion" are required reading for anyone seriously considering creating a new group.*

Remember the first rule in creating new groups: Don't do so unless you are absolutely sure no appropriate group already exists. Usually, you simply haven't found the right group. Once you do, the need to create a group disappears. The Usenet structure lends itself to talking about almost any subject in an existing group. For instance, you can talk about anything Macintosh-related in **comp.sys.mac.misc**. Thus, the second rule of creating a new group: Don't create a group until the traffic in a more general group has grown unmanageable, and stayed that way for some time. As a rule of thumb, wait six months. And, one way or another, make sure you have a Usenet old-timer on your side, to help with the details and steer you clear of any egregious mistakes.

After you are sure the world really does need a group dedicated to discussion of the psychology of smelling flowers from under cork trees, you write a proposal. This *Request for Discussion*, or *RFD*, states what the group is called, what its purpose is, why no existing group serves the need, and so on. Then your job as agitator begins, as you distribute the RFD to groups where interested parties might hang out. Be sure to place the **news.announce.newgroups** group first in the Newsgroups line (so the moderator can correct any problems in the RFD before posting it to **news.announce.newgroups** and the others for you) and to set the Follow-up-To line so that the discussion takes place in **news.groups**. Then you encourage discussion of the topic for 30 days in **news.groups**, all the while collecting responses and modifying the proposal, called a *charter*, accordingly.

After 30 days, if people don't agree on the charter, you must start the RFD process again—with a new and improved proposal, of course. If everyone does agree on the charter, the time has come for a *Call For Votes*, or *CFV*, with clear and unbiased directions on how to vote.

The CFV goes, once again, to all the interested newsgroups, with **news.announce.newgroups** first in the Newsgroups line. It lasts between 21 and 31 days, and you must include the exact end date in the CFV. Once again, your job is to collect and tally the votes via email. (Don't even think of stuffing the electronic ballot box—there's little the Usenet community hates more than a cheat.) You must record each voter's email address along with his Yes or No vote, for later use. You can re-post the CFV during the vote to keep up awareness, but only if you don't change anything from the original CFV.

At the end of the voting period, you post the results—including the total number of votes, and the vote and email address for each—to **news.announce.newgroups** and the other interested newsgroups. Then everyone waits five days, which provides enough time to correct any mistakes or raise serious objections. You need to meet two separate goals to justify a newsgroup. A sufficient number of votes and, within that number, a sufficient number of YES votes. If you have at least 100 more YES votes (for creating the newsgroup) than NO votes, and at least two-thirds of the votes are YES votes, then the group passes the vote.

If, of course, you don't get the required number or percentage of votes, the group doesn't get created. There's no shame in not having your group created. You can even try again in six months if you want; interest may have increased since the original failure. If you fail more than twice, give it up and form your own mailing list. You don't need anyone's cooperation to do that.

If the vote comes out positive, someone (often the moderator of **news.announce.newgroups**) can create the group, sending out the newsgroup control message. Gradually, the group is created at different sites and propagates through much of the network. Why not the entire network? Well, nothing says a machine has to carry every Usenet group in existence. If a system administrator decides that talking about smelling flowers is offensive, she may decide not to carry the group. None of the

machines that rely on her machine for news will have the group, either. Nonetheless, groups focusing on technical issues enjoy relatively complete propagation. Even those discussing topics that some people find offensive enjoy wide propagation, and often greater readership, than the technical groups.

Using Usenet

No matter what software you use to access Usenet, you must be aware of some basic concepts, tasks, and features. When I evaluate different newsreaders such as UUCP/Connect, NewsWatcher, and NewsHopper in later chapters, I tell whether the newsreader in question does a good job of handling these tasks and features for you.

Subscribing to Groups

When you first invoke a newsreader, you must subscribe to the groups you want to read. Occasionally, the newsreader automatically subscribes you to a couple of basic groups, such as **news.newusers.questions** and **news.announce.newusers**. For the most part, however, the thousands of available newsgroups are in the unsubscribed category.

Note

Most machines don't carry all of the Usenet groups. If your machine doesn't carry a group you want, you can either ask the system administrator to get it, or go to a machine that lets anyone read news on it. These sites are called public NNTP sites. Be forewarned, however, that few, if any of these sites still exist, and I know of none. It's a simple problem—whenever a public NNTP site becomes known, so many people try to use it that it immediately stops being public.

Generally, the first time you start up a newsreader it takes a long time because you have to go through all the groups and figure out which ones to subscribe to. The better newsreaders allow you to sort through the list at different times. In the past, you had to sit for an hour or more just unsubscribing from all the groups that you didn't want to read. It was a major hassle. Even now, allot plenty of time to your first session if you're doing it interactively. (Note: This rule doesn't apply to a UUCP connection, where you request only specific groups.)

Reading Articles

After you subscribe to a group, it's time to read the articles. Obviously, the first time you read, all the articles are new to you. After that, you want to make sure that you read only previously unread articles. Most newsreaders are extremely good about keeping track of what you've read already. In the Unix world, the **.newsr**c file tracks what you've read.

Advanced users can edit that file manually with a text editor, to subscribe or unsubscribe from several groups at once. The Macintosh newsreaders make that task, on the whole, unnecessary.

You've learned what the header for a Usenet article might look like, but many newsreaders hide most of the header from you. This is generally helpful, although it can be a pain at times.

Discussions happen in threads, which are groups of related articles, generally with the same or very similar Subject lines. Threads are important because they group both discussions that you want to read and those you don't want to read. Believe me, threads are a very big deal when you have to handle the kind of volume that passes through a popular newsgroup.

When it comes to newsreaders, there are two basic philosophies. The first, which is older, assumes that you want to read 90 percent of the information in a newsgroup. Therefore, the newsreader tries to show you the text of every article unless you explicitly tell it to skip that article or thread. This method may have worked better in the days when Usenet traffic was relatively sparse, but in these modern times, the traffic comes fast and thick. I liken this method to trying to drink from a fire hose.

The second philosophy believes that you want to read only 10 percent of the articles in any given group. With the exception of moderated groups or low-volume groups where every message counts, this assumption is far more realistic. Newsreaders built on this philosophy usually provide a list of the unread messages in a newsgroup, then let you pick and choose which to read. Some newsreaders force you to read each message or thread as you pick it. Others make you pick a whole bunch of them at once and then read them after you've sorted through the entire newsgroup. Both methods have their advantages, and a good newsreader lets you work either way.

Note

*One of the most frequently asked-for programs in **comp.sys.mac.comm** is an off-line newsreader, a program that enables you to save articles to disk and then read them when you aren't connected to the Internet. There are a few ways of getting this capability in a newsreader (UUCP is inherently off-line, for instance), and I discuss them later in this book.*

Navigation

After you start reading a set of messages, you need tools for navigating among them. Navigation tools were more important back in the days when character-based Unix newsreaders were all we had. Today, many of the Macintosh newsreaders replace the navigation commands with mouse actions. However, many people (myself included) find the keyboard to be more efficient than the mouse for navigating through news, so perhaps there's still room for some of the old tools.

The most common navigation capability takes you to the next unread message, whether or not it is in the same thread as the message that preceded it. Closely related is the capability to move to the next unread message in the same thread, even if it's not next to the message you were just reading. In a well-designed newsreader, these two capabilities are closely intertwined, so you don't have to know whether or not you're in a thread.

Note

Often, these navigation features are encapsulated in a single command linked to the Spacebar, which thus serves as an unusual computer command. Essentially, it says to the newsreader, "Do whatever makes the most sense right here." Computers hate those sort of commands, but the concept works extremely well in a newsreader. The Spacebar scrolls down the page. When you hit the bottom of the article, you probably want to read the next article in the thread, so the Spacebar takes you there. When you finish all the articles in that thread, you probably want to go back up and read the next thread, so the Spacebar takes you back up. Finally, after you read everything in a newsgroup, the Spacebar assumes that you want to read the next newsgroup you subscribe to. By making intelligent guesses, a number of commands can be subsumed under that one key.

You want to group discussions into threads so that you can easily read an entire one, even when it spans a fair amount of real world time. You also want to group discussions so you can ignore them more easily. Despite the fact that people should include descriptive Subject lines in their postings, they don't always. If you see a long thread called "Cool Stuff," you have no idea what it's about. It may pique your curiosity, though, so you start reading, only to find out that it's another "my computer is better than your computer" flame war. Now you need a way to kill the entire thread. Good newsreaders make that effort easy.

An even neater feature is the ability to create a list of Subject lines or topics about which you never want to read. This capability usually applies to anything in the header and sometimes to information in the body of the messages, too. It's extremely useful for customizing your Usenet reading experience. To go even one step further, a few newsreaders provide a feature to only read articles that match certain topics. These are ideal for the truly busy user.

After you read all the messages that interest you, it's generally a good idea to mark the rest of them as read (even though you didn't read them). This way, you don't see them again the next time you read news. Some newsreaders handle this option automatically, whereas others make you mark them manually. Sometimes, especially if you just returned from a vacation, you may want to mark everything as read without even trying to read the waiting messages. Marking everything lets you start with a clean slate and with a manageable number of messages the next day, and is generally referred to as *catching up*. There's no difference between a "catch up" feature and a "mark all as read" feature, but you may see both terms.

Now you know all about navigating around Usenet and reading articles. Many people never move past that point, and are called *lurkers*. The term has no negative connotation, it simply means people who only read and never post.

rot13

I almost forgot. You might occasionally run across articles that are completely unreadable. They may be in a newsgroup specific to a language that you don't understand, but the newsreader can't help with that problem. It can (or should be able to) help you with messages coded in the *rot13* format. Rot13 is a simple coding scheme that assigns a number to each letter of the alphabet, starting with 1 for A, 2 for B, and so on, for every character in a message. It then adds 13 to each number and converts back into letters. The result is an utterly unreadable message, which the poster usually intended because some people might find the message offensive. If you see such a message and are easily offended, don't read it. No one forces you to use the rot13 decoding feature that exists in most newsreaders. If you do, you can't very well complain about the contents. I usually see most rot13 encoded postings in joke newsgroups, protecting the innocent from really sick jokes (see figure 8.3).

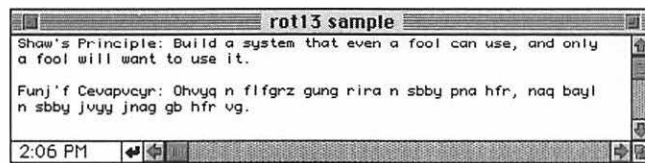


Figure 8.3 Normal text versus rot13.

Extracting Binary Files

In the days of yore, when true Internet connections were less common, files were often passed around the world by being posted in special binaries newsgroups, and even today, you'll see groups like **comp.binaries.mac**. Files are still posted to Usenet, although the majority of them seem to be copyright violations of dirty pictures. A good newsreader makes it easy to download binary files posted in Usenet, although I strongly recommend that you use FTP to get files if you have a choice.

FTP works much better than a newsreader for downloading files and saves a ton of bandwidth because the files aren't being transferred to sites where no one will download them. Of course, if you only have a UUCP connection, or for some other reason you can't use FTP, then snagging binary files posted to Usenet may be your only hope.

Replying to Articles

In the course of reading Usenet news, you often see messages that aren't quite clear or that catch your interest for some reason or another. When you see such a message, you may want to send email to the poster. You could, of course, copy down the poster's email address from the header onto a little piece of paper, and when you finished reading news, use your email program to send him a message. However, that process is a pain and wastes lots of little pieces of paper, so most newsreaders support sending mail replies while you're reading news.

Use email replies whenever the rest of the group won't give a hoot about what you have to say. Most of us feel that our words are pearls of wisdom and should be distributed to the widest possible audience. But, try to step back and think about whether your reply is best directed at the individual making the posting or the group as a whole.

People often ask questions on Usenet, saying that you should reply directly to them and that they plan to summarize to the net. Listen to what these people have to say. They only want replies via email, and because they've promised to post a summary of the replies, you don't need to ask for a copy personally (unless perhaps you don't stand a chance of seeing the summary in the newsgroup; even then, ask nicely). If you ever post a question and promise a summary, live up to your promise, even if you get only a couple of responses. No matter how many responses you get, format them nicely with quote characters before each quoted line so that they are easy for readers to understand; messages are often confusing as to who wrote what in a summary. Never re-post entire headers.

As far as what to say when you reply to postings on Usenet, reread what I said about email manners in the last chapter. The same rules apply here. If you must carry out a flame war, do it in email; but if possible, don't do it at all.

Follow-ups

Discussions are the entire point of Usenet, of course, so you eventually gather the courage to post something to a newsgroup. For most people, the easiest way to post a message is to reply to another message, an action called *following up*. A follow-up is easier for the novice because the newsreader fills in most of the lines in the header for you; the lines for Subject, Newsgroups, Distribution, and so on are generally determined by the message you reply to.

Just as in email, you should be given the chance to quote the previous message so that readers can understand the context of your reply. Some newsreaders are picky about the proportion of quoted text to new text, and for good reason. No one wants to read a two-screen quoted letter only to see at the bottom a few words from you: "I agree with all this." Even in newsreaders that don't prevent you from over-quoting, be careful. Try to edit out as much of the quoted text as possible. Remember that most people have already



seen that message in its original form, so you're simply jogging their memory. Definitely remove signatures and unnecessary previously quoted text.

Note

Using Usenet as a method of getting a message to a specific individual is considered extremely bad form—even if you can't seem to get email to that person. Imagine, everyone's discussing nuclear disarmament, and you suddenly see a message from a college friend. Your note discussing old times at Catatonic University will hold absolutely no interest for the rest of the group. If you find yourself being flamed, suffer and don't do it again.

Posting an Article

If you really have something new to say, or a new question to ask, don't insert it into an existing thread just because it's easier than posting a new one. Posting a new message should be simple enough in any decent newsreader.

Note

*If you cannot post from a newsreader (because you only have an AppleLink account, for instance), you can still send messages to a Usenet newsgroup. Two email posting services exist: **news-group@cs.utexas.edu** and **news-group@pws.bull.com**. Do not send email to **news-group**, but to the name of the group to which you want to post, replacing the dots in the group name with dashes. So, for example, to post to **comp.sys.mac.comm**, send email to **comp-sys-mac-comm@cs.utexas.edu**. Make sure to ask for replies via email.*

In general, you should avoid posting a few things. Avoid copyrighted works such as magazine articles or newspaper stories. Although it's unlikely that anyone could sue the Internet (it would be a bit like boxing with a dense mist), you might be sued for copyright infringement. Besides, posting copyrighted work is not polite. Simply post the complete reference to the article or whatever, along with a summary or selected quote or two if you want to pique some curiosity.

Note

Interestingly, recipes in cookbooks are not copyrighted because they are essentially lists of instructions. However, the instructions for creating the recipe may be protected if they contain anything other than the bare bones instructions, and any preface explaining or describing the recipe is definitely protected. People often post a recipe or two from a cookbook that they particularly like so that others can see whether they like the recipes enough to buy the book.

Perhaps the least obvious but most important works to avoid posting are pictures scanned in from magazines or videos and sounds digitized from TV or videotape. Most of the scanned pictures are varying degrees of erotic images, and unfortunately, most are blatant examples of copyright infringement. The magazines, *Playboy* in particular, don't look kindly on this sort of thing, and legal action might result. Besides, pictures suck up disk space, and the quality of a scanned image doesn't even begin to approach the high-quality photography and printing of most magazines.

In general, you should not post headline events that everyone can read about in the newspaper or possibly in ClariNet (coming up next). I don't mean to imply that you can't talk about these events, but because news travels relatively slowly to all parts of the net, announcing the results of an election or a similar event is just silly. People probably already know about the event, and if they don't, they'll figure it out from the ensuing discussion.

Finally, don't post personal email that you receive unless the sender gives you explicit permission. As with most things on the Internet, posting personal email is a legally murky area, but the etiquette is crystal-clear: It's rude.

ClariNet

Along with all the discussion groups about computers and recreational activities and whatnot, you may see a hierarchy under **clari**. You've found ClariNet. Unlike Usenet, ClariNet doesn't carry any discussions, and in fact, I don't believe that you can post to any ClariNet groups. Instead, ClariNet is dedicated to distributing commercial information, much of it the same stuff that you read in your newspaper or hear on the radio. ClariNet claims over 60,000 daily readers, which isn't too bad in terms of circulation.

Also unlike Usenet, ClariNet isn't free. A site must pay a certain amount to receive the ClariNet news feed, which uses the same transport protocols and newsreaders as Usenet. Sites that receive the ClariNet feed cannot redistribute that feed on to other machines unless those machines pay for it as well. Because of ClariNet's commercial nature, I can't predict whether you even have access to it. It's strictly up to each site.

Much of the ClariNet information comes from press wires like UPI, along with NewsBytes computer articles, and various syndicated columnists such as Miss Manners. A recent arrival is *Dilbert*, the cartoon by Scott Adams (although you have to download each installment and view it in a graphics program, since there isn't a newsreader around that can view graphics internally; although, a properly configured Web browser might be able to handle the task). Although you can probably find much of the information in a standard newspaper, ClariNet organizes it extremely well, making reading about a single topic much easier. For instance, some groups carry local news briefs for each state, some carry only news about Apple Computer, and there are groups with tantalizing names like

clari.news.goodnews, which indeed includes only articles that are good news. (Depressingly, that newsgroup sees very little traffic.)

ClariNet was founded a few years ago by Brad Templeton, who is also well known as the creator of the moderated group **rec.humor.funny**, which accepts only jokes that he thinks are funny (actually, someone else does the selection now). ClariNet is important, because it is specifically commercial traffic flowing via the same methods and pathways as Usenet, perhaps the most rabidly anti-commercial part of the Internet. I don't know the business details of ClariNet, but it has been around long enough that I suspect it's a financial success, and the news that it brings is certainly welcome.

Enough Usenet

Well, that's enough on Usenet. You must experience Usenet to truly understand it, though, so I do recommend that you find a few groups that interest you and lurk for a while once you have your connection set up. You're now ready to learn about the services that require a full connection to the Internet, such as FTP, Telnet, WAIS, Gopher, and the World Wide Web.

Chapter 9



TCP/IP Internet Services

I must tread a fine line when talking about Internet services, because the level of connection (and thus the level of service) varies widely. People who can send Internet email, for instance, may not be able to use Gopher or the World Wide Web. The services that I talk about in this chapter (except for FTP and Archie via email) all require a full TCP/IP connection to the Internet. For Mac folks, a full TCP/IP connection to the Internet means that you have MacTCP loaded and properly configured, and either a dedicated Internet connection or a modem connection via PPP or SLIP. If you don't have the proper sort of account and connection, you do not have a full TCP/IP connection to the Internet. To be fair, there are some ways of getting access to these services via America Online, CompuServe, or some of the bulletin board systems, and in those cases you're limited to the software they provide for you.

FTP

Despite the occasionally confusing way people use the term both as a noun and a verb, most people don't have much trouble using FTP. FTP stands for *File Transfer Protocol*, and not surprisingly, it's only good for transferring files between machines. In the past, you could only use an FTP client to access files stored on FTP servers. Today, however,

enough other services such as Gopher and the World Wide Web have implemented the FTP protocols that you can often FTP files no matter what service you happen to be using. Heck, you can even FTP files via email. I'll get to the specifics of the different clients in later chapters; for now, here are a few salient facts to keep in mind regarding FTP.

FTP Manners

The Internet does a wonderful job of hiding geographical boundaries. You may never realize that a person with whom you correspond lives on the other side of the globe. When using FTP, however, try to keep the physical location of the remote machine in mind.

First, as with everything on the Internet, someone pays for all this traffic. It's probably not you directly, so try to behave like a good citizen who's being given free access to an amazing resource. Show some consideration by not, for example, using machines in Australia when one in the same area of your country works equally well. Because trans-oceanic traffic is expensive, many machines mirror others; that is, they make sure to transfer the entire contents of one machine to the other, updating the file collection on a regular, often daily basis.

Here's an example. Because the Info-Mac archive site at **sumex-aim.stanford.edu** is popular and kept up-to-date, other sites carrying Macintosh software don't want to duplicate the effort. It's much easier to set up a mirror to **sumex** so that machines in Australia and Scandinavia can have exactly the same contents as **sumex**. Mirroring not only saves work, it also enables users in those countries to access a cheaper, local site for files. Everyone wins, but only if you, the user, utilize local sites whenever possible. You can usually tell where a site is located by looking at the two-letter country domain at the end of the address.

Sometimes, of course, the file you need exists only on a remote site in Finland, for example, so that's where you must go to get it. Another point of etiquette to keep in mind is sensitivity to the time of day at the site from which you retrieve a file. Like most things in life other than universities during exams, more people use the Internet during their daytime hours than at night. Thus, it's generally polite to retrieve files during off hours; otherwise, you're preventing people from doing their work. That's not polite, especially if the file you're retrieving is a massive QuickTime movie or something equally frivolous.

Notice that I said "their daytime hours." Because the Internet spans the globe, it may be 4:00 A.M. where you are, but it's the middle of the day somewhere else. You can figure out the local time by using the Map control panel that comes with your Mac.

FTP Clients

FTP is inherently simple to use, but there's plenty of room for FTP client software to make your life miserable. The following sections, therefore, describe several benefits and features to look for in an FTP client.

Connecting

People usually use an FTP client program to log on to a remote FTP site, find a file or two, download them, and then log off. As such, a disproportionate amount of your time is spent connecting and navigating to the desired files.

A good FTP client enables you to define shortcuts for frequently used FTP sites, along with the userid and password necessary for connecting to them. This benefit is minor but makes a big difference when repeated numerous times. I can't tell you how much I hate typing **sumex-aim.stanford.edu** on a Unix command line when I'm trying to connect to that site with FTP.

Navigating

Once you're on, the FTP client program should make it very easy to move between directories (or folders, in Mac jargon). Some programs do this by emulating the Standard File Dialog used on the Mac to open and save files, which is a good start (although the Standard File Dialog is one of the most confusing parts of the Macintosh interface). It's helpful when the client program remembers the contents of directories. That way, if you go back to one you've already visited, you don't have to wait for it to refresh.

Other programs, Anarchie and Snatcher mostly, take the navigational aspect of FTP one step further, and actually emulate the Finder. Snatcher in particular goes a bit overboard in trying to mimic the Finder, in my opinion.

A useful variant of shortcuts (also known as bookmarks) to FTP site names is the addition of directory information to the site name. Say, for instance, you want to retrieve something from **ftp.tidbits.com**. Not only do you have to enter the host name, userid, and password, but you must also go to the proper directory, which is **/pub/tidbits**.



Listing Style

In Unix, you can choose among several different methods of viewing files. Some show you more information, such as file size and creation date, and others show you less, in order to fit more entries on the screen. Although the Mac doesn't have the problem of trying to fit multiple columns in a list (no Macintosh program uses multiple column lists), not all the FTP clients are good about showing you the entire filename, size, or date. I

think this failure is inexcusable, because you need to know how large a file is before you spend an hour retrieving it—especially if you’re connecting at a slow speed. Make sure the program you use provides this information. In addition, a truly useful feature is the capability of sorting the listing on date, file size, or whatnot.

Recognizing File Type and Decoding

Much of the time, an FTP client can figure out what sort of file you’re retrieving by looking at the extension to the filename. This being the case, the client can make sure it is transferring the file in the proper format. If you’re lucky, it even decodes some of the common formats that you see on the Internet.

“Wait a minute,” you say. “He didn’t mention strange file formats before.” Sorry about that. I’ll get to file formats in the next chapter, after I’ve discussed the various different ways that files may appear on your machine. Let’s talk about how you find files via FTP. The answer is Archie.

Archie

Archie is an example of what happens when you apply simple technology to a difficult problem. Here is the problem: How do you find any given file on the nets if you don’t already know where it’s located? After all, in comparison with finding a single file on several million machines, the proverbial haystack looks tiny, and its cousin, the proverbial needle, sticks out like the sore thumb you get when you find it. In a nutshell, Archie uses normal FTP commands to get directory listings of all the files on hundreds of anonymous FTP sites around the world. It then puts these file listings into a database and provides a simple interface for searching it. That’s really all there is to Archie.

Unfortunately, and for reasons I don’t fully understand, Archie servers have become less and less useful over time. They’re almost impossible to get through to via an Archie client (telnetting to them is the most successful in my recent experience), and much of the time they don’t seem to know about certain large FTP sites that I know have the file for which I’m looking. In other words, sometimes Archie simply won’t work. Don’t worry about it and just try another technique or tool.

Note

Archie was developed in early 1991 by Alan Emtage, Peter Deutsch, and Bill Heelan from the McGill University Computing Center, Canada. Development now takes place at a company founded by Deutsch and Emtage, Bunyip Information Systems. Although the basic Archie client software is distributed freely, Bunyip sells and supports the Archie server software.

You can access Archie via Telnet, email, Gopher, the World Wide Web, and special Macintosh client programs. Some Unix machines may also have Unix Archie clients

installed. It seems to me there are two basic goals an Archie client should meet. First, it should be easy to search for files, but when you want to define a more complex search, that should be possible as well. Second, since the entire point of finding files is so that you can retrieve them, an Archie client ideally should make it very easy to retrieve anything that it finds. This second feature appears to be less common than you would expect. On the Mac, only Anarchie can retrieve found files with just a double-click.

Note

Archie isn't an acronym for anything, although it took me half an hour searching through files about Archie on the Internet to determine that once and for all.

Accessing Archie via email is extremely easy, although the Archie server offers enough options (I'll let you discover them for yourself) to significantly increase the complexity. For a basic search through, merely send email to **archie@archie.internic.net** and put in the body of the message lines like the following:

```
help
find easy-view
find easyview
```

In a short while (or perhaps a long while, depending on the load on the Archie server), the results should come back—the help file that you asked for and the results of your search for “easy-view” and “easyview.” The example above uses both terms because I’m not sure of the exact wording of the filename, but experience tells me that one of those two possibilities is likely.

However, if the Archie server you chose is down, or merely being flaky (as is their wont) you may want to try another one. There are plenty. Simply send email to the userid **archie** at any one of the Archie servers from the list in Table 9.3. As usual, it’s polite to choose a local server.

Table 9.3
Current Archie Servers

Server Name	Server IP Number	Location
archie.au	139.130.4.6	Australia
archie.edvz.uni-linz.ac.at	140.78.3.8	Austria
archie.univie.ac.at	131.130.1.23	Austria

continues

Table 9.3

continued

<i>Server Name</i>	<i>Server IP Number</i>	<i>Location</i>
archie.cs.mcgill.ca	132.206.51.250	Canada
archie.uqam.ca	132.208.250.10	Canada
archie.funet.fi	128.214.6.102	Finland
archie.univ-rennes1.fr	129.20.128.38	France
archie.th-darmstadt.de	130.83.128.118	Germany
archie.ac.il	132.65.16.18	Israel
archie.unipi.it	131.114.21.10	Italy
archie.wide.ad.jp	133.4.3.6	Japan
archie.hana.nm.kr	128.134.1.1	Korea
archie.sogang.ac.kr	163.239.1.11	Korea
archie.uninett.no	128.39.2.20	Norway
archie.rediris.es	130.206.1.2	Spain
archie.luth.se	130.240.12.30	Sweden
archie.switch.ch	130.59.1.40	Switzerland
archie.ncu.edu.tw	192.83.166.12	Taiwan
archie.doc.ic.ac.uk	146.169.11.3	United Kingdom
archie.hensa.ac.uk	129.12.21.25	United Kingdom
archie.unl.edu	129.93.1.14	USA (NE)
archie.internic.net	198.49.45.10	USA (NJ)
archie.rutgers.edu	128.6.18.15	USA (NJ)
archie.ans.net	147.225.1.10	USA (NY)
archie.sura.net	128.167.254.179	USA (MD)

Telnet Usage

Telnet is a bit hard to talk about because using it is just like using a modem to connect to another computer. Telnet simply enables you to connect to a computer somewhere else on the Internet and to do whatever that computer allows you to do. Because Telnet is similar to FTP in the sense that you're logging in to a remote machine, the same rules of etiquette apply (although running a program over Telnet usually places less stress on a machine).

As long as you try to avoid bogging down the network when people want to use it for their local work, you shouldn't have to worry about it too much. When you telnet to another machine, you generally telnet into a specific program that provides information you want. The folks making that information available may have specific restrictions on the way you can use their site. Pay attention to these restrictions. The few people who abuse a network service ruin it for everyone else.

What might you want to look for in a Telnet program? That's a good question, I suppose, but not one that I'm all that qualified to answer. For the most part, I avoid Telnet-based command-line interfaces. Thus, in my opinion, you should look for features in a Telnet program that will make using it, and any random program that you may happen to run on the remote machine, easier to use.

It's useful to be able to save connection documents that save you the work of logging into a specific machine (but beware of security issues if they also store your password). Also, any sort of macro capability will come in handy for automating repetitive keystrokes. Depending on what you're doing, you also may want some feature for capturing the text that flows by for future reference. And, you should of course be able to copy and paste out of the Telnet program.

IRC

IRC, which stands for *Internet Relay Chat*, is a method of communicating with others on the Internet in real time. It was written by Jarkko Oikarinen of Finland in 1988 and has spread to 20 countries. IRC is perhaps better defined as a multi-user chat system, where people gather in groups that are called *channels*, usually devoted to some specific subject. Private conversations also are possible.

Note

IRC gained a certain level of fame during the Gulf War, when updates about the fighting flowed into a single channel where a huge number of people had gathered to stay up-to-date on the situation.

I personally have never messed with IRC much, having had some boring experiences with RELAY, a similar service on BITNET, back in college. I'm not all that fond of IRC, in large part because I find the amount of useful information there almost nonexistent, and I'm uninterested in making small talk with people from around the world. Nevertheless, IRC is one of the most popular Internet services. Thousands of people connect to IRC servers throughout any given day. You can find more information in the IRC tutorials posted for anonymous FTP in:

<ftp://cs-ftp.bu.edu/irc/support/>

Client programs for many different platforms exist, including two for the Macintosh called *ircle* and *Homer*. Much as with Telnet, you're looking for features that make the tedious parts of IRC simpler. I could blather on about all the features you might want, but frankly, if you're using a Macintosh with either a Unix shell account or a MacTCP-based account, just get *Homer*. It has more features than one would think possible, and can even—in conjunction with Apple's PlainTalk software—speak some or all of the text that flows by.

MUDs

MUD, which stands for *Multi-User Dungeon* or often *Multi-User Dimension*, may be one of the most dangerously addictive services available on the Internet. The basic idea is somewhat like the text adventures of old, where you type in commands like "Go south," "Get knife," and so on. The difference with MUDs is that they can take place in a wide variety of different realities—basically anything someone could dream up. More importantly, the characters in the MUD are actually other people interacting with you in real time. Finally, after you reach a certain level of proficiency, you are often allowed to modify the environment of the MUD.

The allure of the MUDs should be obvious. Suddenly, you can become your favorite alter-ego, describing yourself in any way you want. Your alternate-reality prowess is based on your intellect, and if you rise high enough, you can literally change your world. Particularly for those who may feel powerless or put upon in the real world, the world of the MUD is an attractive escape, despite its text-environment limitations.

After the publication of an article about MUDs, the magazine *Wired* printed a letter from someone who had watched his brother fail out of an engineering degree and was watching his fiancée, a fourth-year astrophysics student, suffer similar academic problems, both due to their addictions to MUDs. But don't take my word for it; read the letter for yourself on *Wired's* Web server:

<http://www.wired.com/Etext/1.4/departments/rants.html>

Note

Wired's Web server requires authentication now, which means that you must sign up with them and get a userid and a password before you can get in. It's free, and you can register at:

<http://www.wired.com/>

I've seen people close to me fall prey to the addictive lure of MUDs. As an experiment in interactive communications and human online interactions, MUDs are extremely interesting, but be aware of the time they can consume from your real life.

I don't want to imply that MUDs are evil. Like almost anything else, they can be abused. But in other situations, they have been used in fascinating ways, such as to create an online classroom for geographically separated students. There's also a very real question of what constitutes addiction and what constitutes real life. I'd say that someone who is failing out of college or failing to perform acceptably at work because of a MUD has a problem, but if that person is replacing several hours per day of television with MUDing, it's a tougher call. Similarly, is playing hours and hours of golf each week any better than stretching your mind in the imaginative world of a MUD? You decide, but remember: there are certain parts of real life that we cannot and should not blow off in favor of a virtual environment.

Although MUDs are currently text-only, rudimentary graphics will almost certainly appear at some point, followed by more realistic graphics, sound, and video, and perhaps some day even links to the virtual reality systems of tomorrow. I don't even want to speculate on what those changes might mean to society, but you may want to think about what may happen, both positive and negative.

MUDs generally run under Unix, but you could run your own with a Macintosh port of a MUD, called MacMud, and connect to other Unix MUDs with a simple MUD client program, MUDDweller. Even more interesting is the program Meeting Space from a small company called World Benders. Meeting Space is billed as a virtual conference room, and is marketed to large businesses as a money- and time-saving alternative to business trips. However, it's actually a business MUD with a snazzy Macintosh interface and hefty price tag. Meeting Space works over any Macintosh network, including the Internet, and although I don't know of any public Meeting Space servers yet, some were being discussed earlier. For more information about Meeting Space, send email to **wb-info@worldbenders.com**.

WAIS

Unlike almost every other resource mentioned in this book, the *WAIS*, or *Wide Area Information Servers*, project had its conception in big business and was designed for big business. The project started in response to a basic problem. Professionals from all walks of life, and corporate executives in particular, need tremendous amounts of information that is usually stored online in vast databases. However, corporate executives are almost always incredibly busy people without the time, inclination, or skills to learn a complex database query language. Of course, corporate executives are not alone in this situation; many people have the same needs and limitations.

In 1991, four large companies—Apple Computer, Dow Jones & Co., Thinking Machines Corporation, and KPMG Peat Marwick—joined together to create a prototype system to address this pressing problem. Apple brought its user interface design expertise; Dow Jones was involved because of its massive databases of information; Thinking Machines provided the programming and expertise in high-end information retrieval engines; and KPMG Peat Marwick provided the information-hungry guinea pigs.



One of the initial concepts was the formation of an organizational memory—the combined set of memos, reports, guidelines, email, and whatnot—that make up the textual history of an organization. Because all of these items are primarily text and completely without structure, stuffing them into a standard relational database is like trying to fill a room with balloons. They don't fit well, they're always escaping, and you can never find anything. WAIS was designed to help with this problem.

So far I haven't said anything about how WAIS became such a useful tool for finding free information. With such corporate parentage, it's in some ways surprising that it did. The important thing about the design of WAIS is that it doesn't discriminate. WAIS can incorporate data from many different sources, distribute them over various types of networks, and record whether the data is free or carries a fee. WAIS is also scalable, so that it can accept an increasing number and complexity of information sources. This is an important feature in today's world of exponentially increasing amounts of information. The end result of these design features is that WAIS works perfectly well for serving financial reports to harried executives, but equally well for providing science fiction book reviews to curious undergraduates.

In addition, the WAIS protocol is an Internet standard and is freely available, as are some clients and servers. Anyone can set up her own WAIS server for anyone with a WAIS client to access. Eventually, we may see Microsoft, Lotus, and WordPerfect duking it out over who has the best client for accessing WAIS. With the turn the Internet has taken in the past year, however, it's far more likely that we'll see Microsoft, Lotus, and WordPerfect (now a division of Novell) competing with World Wide Web clients. Although WAIS has continued to grow in utility and popularity, it has also faded into the shadow of the snazzier looking Web clients. That's not to say that WAIS isn't being used heavily, just that it tends to work behind the scenes as a search engine for a Web page interface, rather than through a dedicated client program.

At the beginning of this section, I mentioned the problem of most people not knowing how to communicate in complex database query languages. WAIS solves that problem by implementing a sophisticated natural language input system, which is a fancy way of saying that you can talk to it in English. If you want to find more information about deforestation in the Amazon rain forest, you simply formulate your query as: "Tell me about deforestation in the Amazon rain forest." Pretty rough, eh? In its current state, WAIS does not actually examine your question for semantic content; that is, it searches based on the useful words it finds in your question (and ignores, for instance, "in" and "the"). However, nothing prevents advances in language processing from augmenting WAIS so that it has a better idea of what you mean.

In any database, you find only the items that match your search. In a very large database, though, you often find far too many items; so many, in fact, that you are equally at a loss as to what might be useful. WAIS attempts to solve this problem with *ranking* and *relevance feedback*. Ranking is just what it says. WAIS looks at each item that answers the

user's question and ranks them based on the proximity of words and other variables. The better the match, the higher up the document appears in your list of found items. Although by no means perfect, this basic method works well in practice.

Relevance feedback, although a fuzzier concept, also helps you refine a search. If you ask a question and WAIS returns 30 documents that match, you may find one or two that are almost exactly what you're looking for. You can then refine the search by telling WAIS, in effect, that those one or two documents are "relevant" and that it should go look for other documents that are "similar" to the relevant ones. Relevance feedback is basically a computer method of pointing at something and saying, "Get me more like this."

The rise of services such as WAIS and Gopher on the Internet will by no means put librarians out of business. Instead, the opposite is true. Librarians are trained in ways of searching and refining searches. We need their experience, both in making sense of the frantic increase in information resources and in setting up the information services of tomorrow. More than ever, we need to eliminate the stereotype of the little old lady among dusty books and replace it with an image of a person who can help us navigate through data in ways we never could ourselves. There will always be a need for human experts.

When you put all this information together, you end up with a true *electronic publishing system*. This definition, pulled from a paper written by Brewster Kahle, then of Thinking Machines and now president of WAIS, Inc., is important for Internet users to keep in mind as the future becomes the present: "Electronic publishing is the distribution of textual information over electronic networks." (Kahle later mentions that the WAIS protocol does not prohibit the transmission of audio or video.) I emphasize that definition because I've been fighting to spread it for some years now because of my role with *TidBITS*.

Note

Electronic publishing has little to do with using computer tools to create paper publications. For those of you who know about Adobe Acrobat, Common Ground from No Hands Software, Envoy from Novell, and Replica from Farallon, those programs aren't directly related to electronic publishing because they all work on the metaphor of a printed page. With them, you create a page and then print to a file format that other platforms can read (using special readers), but never edit or reuse in any significant way. We're talking about electronic fax machines. We should enjoy greater flexibility with electronic data.

So, how can you use WAIS? I see two basic uses. Most of the queries WAIS gets are probably one-time shots where the user has a question and wants to see whether WAIS stores any information that can provide the answer. This use has much in common with the way reference librarians work—someone comes in, asks a question, gets an answer, and leaves.

More interesting for the future of electronic publishing is a second use, that of periodic information requests. As I said earlier in this book, most people read specific sections of the newspaper and, even within those sections, are choosy about what they do and don't read. I, for instance, always read the sports section but I am interested only in baseball, basketball, football to a lesser extent, and hockey only if the Pittsburgh Penguins are mentioned. Even within the sports I follow closely, baseball and basketball, I'm more interested in certain teams and players than others.

Rather than skim through the paper each Sunday to see whether anything interesting happened to the teams or players I follow, I can instead ask a question of a WAIS-based newspaper system (which is conceivable right now, using the UPI news feed that ClariNet sells via Usenet). In fact, I might not ask only one question, but I may gradually come up with a set of questions, some specific, others abstract. Along with "What's happening with Cal Ripken and the Baltimore Orioles?" could be "Tell me about the U.S. economy."

In either case, WAIS would run my requests periodically, every day or two, and indicate which items are new in the list. Ideally, the actual searching would take place at night to minimize the load on the network and to make the search seem faster than the technology permits. Once again, this capability is entirely possible today; all that lacks for common usage is the vast quantities of information necessary to address everyone's varied interests. Although the amount of data available in WAIS is still limited (if you call 700-plus sources limited), serious and important uses are already occurring.

In large part due to its corporate parentage, the WAIS project has been careful to allow for information to be sold and for owners of the information to control who can access the data and when. Although not foolproof, the fact that WAIS addresses these issues makes it easier to deal with copyright laws and information theft.

Because of the controls WAIS allows, information providers are likely to start making sources of information more widely available. With the proliferation of these information sources, it will become harder for the user to keep track of what's available. To handle that problem, WAIS incorporates a Directory of Servers, which tracks all the available information servers. Posing a question to the Directory of Servers source (WAIS calls sets of information *sources* or *servers*) returns a list of servers that may have information pertaining to your question. You can then easily ask the same question of those servers to reach the actual data.

Most of the data available on WAIS is public and free at the moment, and I don't expect that arrangement to change. I do expect more commercial data to appear in the future, however.

In regard to that issue I want to propose two ideas. First, charges should be very low to allow and encourage access, which means that profit is made on high volume rather than high price. Given the size of the Internet, I think this approach is the way to go, rather than charging exorbitant amounts for a simple search that may not even turn up the answer to your question.

Second, I'd like to see the appearance of more "information handlers," who foot the cost of putting a machine on the Internet and buying WAIS server software and then, for a percentage, allow others to create information sources on their server. WAIS, Inc. already provides this service, but I haven't heard of much competition yet. That service enables a small publisher to make, say, a financial newsletter available to the Internet public for a small fee, but the publisher doesn't have to go to the expense of setting up and maintaining a WAIS server. This arrangement will become more commonplace; the question is when? Of course, as the prices of server machines, server software, and network connections drop, the number of such providers will increase.

WAIS has numerous client interfaces for numerous platforms, but you probably can use either a simple VT100 interface via Telnet or, if you have a MacTCP link to the Internet, a program called MacWAIS. When evaluating WAIS client programs, keep in mind my comments about the two types of questions and the relevance feedback. A WAIS client should make it easy to ask a quick question without screwing around with a weird interface, and it should also enable you to save questions for repeated use (as in the electronic newspaper example). Similarly, with relevance feedback, that act of pointing and saying, "Find me more like this one that I'm pointing at" should be as simple as possible without making you jump through hoops.

Finally, none of the WAIS clients I've seen provide a simple method of keeping track of new sources as they appear, not to mention keeping track of which sources have gone away for good.

Gopher

In direct contrast to WAIS, Gopher originated in academia at the University of Minnesota, where it was intended to help distribute campus information to staff and students. The name is actually a two-way pun (there's probably a word for that) because Gopher was designed to enable you to "go fer" some information. Many people probably picked up on that pun, but the less well-known one is that the University of Minnesota is colloquially known as the home of the Golden Gophers, the school mascot. In addition, one of the Gopher Team members said that there are gophers living outside their office.

Note

Calling yourself the Golden Gophers makes more sense than calling yourself the Trojans, not only considering that the Trojans were one of the most well-known groups in history that lost, but also considering that they lost the Trojan War because they fell for a really dumb trick. "Hey, there's a gigantic wooden horse outside, and all the Greeks have left. Let's bring it inside!" Not a formula for long-term survival. Now, if they had formed a task force to study the Trojan Horse and report back to a committee, everyone wouldn't have been massacred. Who says middle management is useless? Anyway, I digress.

The point of Gopher is to make information available over the network, much in the same way that FTP does. In some respects, Gopher and FTP are competing standards for information retrieval, although they serve somewhat different purposes. Gopher only works for retrieving data; you cannot use it to send data. Also, there's no easy way to give Gopher users usernames and passwords so only they can access a Gopher site.

Gopher has several advantages over FTP. First, it provides direct access to far more types of information resources than FTP. Gopher provides access to online phone books, online library catalogs, the text of the actual files, databases of information stored in WAIS, various email directories, Usenet news, and Archie. Second, Gopher pulls all this information together under one interface and makes it all available from a basic menu system.

Note

Menu items on a Gopher server are not Macintosh menus, but list items in a Macintosh window under TurboGopher. Keep that in mind, and you'll be fine.

If you retrieve a file via FTP and the file gives you a reference to another FTP server, you as the user must connect to that site separately to retrieve any more files from there. In contrast, you connect to a single home Gopher server, and from there, wend your way out into the wide world of Gopherspace without ever having to consciously disconnect from one site and connect to another (although that is what happens under the hood). Gopher servers almost always point at each other, so after browsing through one Gopher server in Europe, you may pick a menu item that brings you back to a directory on your home server. Physical location matters little, if at all, in Gopherspace.

Gopher has also become popular because it uses less net bandwidth than standard FTP. When you connect to a Gopher server, the Gopher client software actually connects only long enough to retrieve the menu, and then it disconnects. When you select something from the menu, the client connects again very quickly, so you barely notice that you weren't actually wasting net bandwidth during that time. Administrators like using Gopher for this reason. They don't have to use as much computing power providing files to Internet users.

Note

There's actually no reason why FTP servers couldn't be rewritten to work this way, as well. Jim Matthews, the author of Fetch, is always going on about how writing an FTP server that used something called lightweight threads would make FTP more efficient. In the meantime, Peter Lewis's Anarchie FTP client for the Mac works much like a Gopher client in that it is continually connecting again and again to your target FTP site, enabling you to perform more than one FTP task at a time.

Several Gopher clients exist for the Macintosh. The one written by the Gopher programmers themselves is arguably the best Gopher client for any platform. They claim that it's the fastest over slow connections, and although I haven't used clients on other platforms, TurboGopher is certainly fast. You also can access Gopher via Telnet and a VT100 interface. It's nowhere near as nice (it's slower, you can only do one thing at a time, and you cannot view pictures and the like online), but it works if you don't have MacTCP-based access to the Internet.

Veronica

The most important adjunct to Gopher is a service called Veronica, developed by Steve Foster and Fred Barrie at University of Nevada. Basically, Veronica is to Gopher what Archie is to FTP—a searching agent; hence, the name.

Note

Veronica stands for Very Easy Rodent-Oriented Net-wide Index to Computerized Archives, but apparently the acronym followed the name.

Veronica servers work much like Archie servers. They tunnel through Gopherspace recording the names of available items and adding them to a massive database.

You usually find a Veronica menu within an item called Other Gopher and Information Servers, or occasionally simply World. When you perform a Veronica search, you either look for Gopher directories, which contain files, or you look for everything available via Gopher, which includes the files and things like WAIS sources as well. There are only a few Veronica servers in the world (between four and six, depending on which machines are up), so you may find that the servers are heavily overloaded at times, at which point they'll tell you that there are too many connections and that you should try again later. Although it's not as polite as I'd like, I find that using the European Veronica servers during their night is the least frustrating.

It's definitely worth reading the "Frequently Asked Questions about Veronica" document that lives with the actual Veronica servers. It provides all sorts of useful information about how Veronica works, including the options for limiting your search to only directories or only searchable items. You can use Boolean searches within Veronica, and there are ways of searching for word stems—that is, the beginning of words. So, if you wanted to learn about yachting, you could search for "yacht*." The possibilities aren't endless, but Veronica is utterly indispensable for navigating Gopherspace and for searching on the Internet in general.

Jughead

Getting sick of the Archie Comics puns yet? They just keep coming and, like Veronica, I somehow doubt that this acronym came before the name. *Jughead* stands for *Jonzy's Universal Gopher Hierarchy Excavation And Display*. Jughead does approximately the same thing as Veronica, but if you've ever done a Veronica search on some generic word, you know that Veronica can provide just a few too many responses (insert sarcasm here). Jughead is generally used to limit the range of a search to a certain machine, and to limit it to directory titles. This makes Jughead much more useful than Veronica if you know where you want to search, or if you're only searching on a Gopher server that runs Jughead.

I don't use Jughead all that much, because what I like about the massive number of Veronica results is that they often give me a sense of what information may exist on any given topic. I suppose that if I regularly performed fairly specific searches on the same set of Gopher servers, I'd use Jughead more.

Note

The best way to find a generally accessible Jughead server is to do a Veronica search on "jughead -t7." That returns a list of all searchable Jughead servers, rather than all the documents and directories in Gopherspace that contain the word "jughead."

World Wide Web

The World Wide Web is the most recent and ambitious of the major Internet services. The Web was started at CERN, a high-energy physics research center in Switzerland, as an academic project. It attempts to provide access to the widest range of information by linking not only documents made available via its native *HTTP (HyperText Transfer Protocol)*, but also additional sources of information via Usenet news, FTP, WAIS, and Gopher. The Web tries to suck in all sorts of data from all sorts of sources, avoiding the problems of incompatibility by allowing a smart server and a smart client program to negotiate the format of the data.

Note

CERN doesn't stand for anything any more, but it once was an acronym for a French name.

In theory, this capability to negotiate formats enables the Web to accept any type of data, including multimedia formats, once the proper translation code is added to the servers and the clients. And, when clients don't understand the type of data that's appearing,

such as a QuickTime movie, for instance, they generally just treat the data as a generic file, and ask another program to handle it after downloading.

The theory behind the Web makes possible many things, such as linking into massive databases without the modification of the format in which they're stored, thereby reducing the amount of redundant or outdated information stored on the nets. It also enables the use of intelligent agents for traversing the Web. But what the Web really does for the Internet is take us one step further toward total ease of use. Let's think about this evolution for a minute.

FTP simply transfers a file from one place to another—it's essentially the same thing as copying a file from one disk to another on the Mac. WAIS took the concept of moving information from one place to another, and made it possible for client and server to agree on exactly what information is transferred. When that information is searched or transferred, you get the full text without having to use additional tools to handle the information. Gopher merged both of those concepts, adding in a simple menu-based interface that greatly eased the task of browsing through information. Gopher also pioneered the concept of a virtual space, if you will, where any menu item on a Gopher server can refer to an actual file anywhere on the Internet. Finally, the World Wide Web subsumes all of the previous services and concepts, so it can copy files from one place to another; it can search through and transfer the text present in those files; and it can present the user with a simple interface for browsing through linked information.

But aside from doing everything that was already possible, the World Wide Web introduced four new concepts. The first one I've mentioned already—it's the capability to accept and distribute data from any source, given an appropriately written Web server.

Second, the Web introduced the concept of rich text and multimedia elements in Internet documents. Gopher and WAIS can display the text in a document, but they can't display it with fonts and styles and sizes and sophisticated formatting. You're limited to straight, boring text (not that it was boring when it first appeared, I assure you). With the Web, you can create *HTML* documents that contain special codes that tell a Web browser program to display the text in various different fonts and styles and sizes. *Web pages* (that's what documents on the Web are generally called) also can contain *inline graphics*—that is, graphics that are mixed right in with the text, much as you're used to seeing in books and magazines. And finally, for something you're not used to seeing in books and magazines, a Web page can contain sounds and movies, although sound and movie files are so large that you must follow a link to play each one.

Link? What's a link? Ah, that's the third concept that the Web brought to the Internet. Just as an item in a Gopher menu can point to a file on another Internet machine in a different country, so can Web *links*. The difference is that any Web page can have a large number of links, all pointing to different files on different machines, and those links can be embedded in the text. For instance, if I were to say in a Web page that I have a really great collection of **penguin pictures** stored on another Web page (and if you were reading

this on the Web and not in a book), you could simply click on the underlined words to immediately jump to that link. Hypertext arrives on the Internet.

Hmm, I should probably explain hypertext. A term coined by Ted Nelson many years ago, *hypertext* refers to nonlinear text. Whereas you normally read left to right, top to bottom, and beginning to end, in hypertext you follow links that take you to various different places in the document, or even to other related documents, without having to scan through the entire text. Assume, for instance, that you're reading about wine. There's a link to information on the cork trees that produce the corks for wine bottles, so you take that link, only to see another link to the children's story about Ferdinand the Bull, who liked lying under a cork tree and smelling the flowers. That section is in turn linked to a newspaper article about the running of the bulls in Pamplona, Spain. A hypertext jump from there takes you to a biography of Ernest Hemingway, who was a great fan of bull fighting (and of wine, to bring us full circle). This example is somewhat facetious, but hopefully it gives you an idea of the flexibility a hypertext system with sufficient information, such as the World Wide Web, can provide.

Fourth, the final new concept the Web introduced to the Internet is *forms*. Forms are just what you would think, online forms that you can fill in, but on the Internet, forms become tremendously powerful because they make possible all sorts of applications, ranging from surveys to online ordering to reservations to searching agents to who knows what. Forms are extremely useful, and are increasingly heavily used on the Web for gathering information in numerous contexts.

For some time, the Web lacked a searching agent such as Archie or Veronica, a major limitation because the Web is so huge. However, a number of searching agents have appeared, and although they simply don't feel as successful as Veronica yet, I suspect that's merely because I'm less used to them. You can find a page of the Web searching agents at:

<http://cuiwww.unige.ch/meta-index.html>

In addition, a number of useful subject catalogs have sprung up; currently my favorite one is called Yahoo, and can be accessed at:

<http://www.yahoo.com/>

You can access the Web via a terminal and a VT100 interface, or even via email (which is agonizingly slow), but for proper usage, you must have a special browser.

NOTE

To try the Web via email, send email to **listproc@www0.cern.ch** with the command **www** in the body of the message.

When you're evaluating Web browsers, there are a number of features to seek. The most important is one that seems obvious: an easy way to traverse links. Because the entire point of a Web browser is to display fonts and styles in text, a Web browser should give you the ability to change the fonts to ones on your Mac that you find easy to read. HTML documents don't actually include references to Times and Helvetica; they encode text in certain styles, much like a word processor or page layout program does. Then, when your Web browser reads the text of a Web page, it decodes the HTML styles and displays them according to the fonts that are available. Sometimes the defaults are ugly, so I recommend playing with them a bit. Many, if not most, Web pages also contain graphics, which is all fine and nice unless you're the impatient sort who dislikes waiting for the graphics to travel over a slow modem. Web browsers should have an option to turn off auto-loading of images or let you move on before the images have finished loading. You should be able to do anything you can do in a normal Mac application, such as copy and paste. You should be able to save a hotlist, preferably hierarchical, of Web sites that you'd like to visit again. Finally, you should be able to easily go back to previously visited pages without having to reload them over the Internet.

As I said previously, there are a number of ways to access the Web. But frankly, if you use a Mac and don't have access to a MacTCP-based connection, you'll miss out on the best parts, even if you can see the textual data in a VT100 browser such as Lynx.

Wrapping Up

That should do it for the background material about the various TCP/IP Internet services, such as FTP, Telnet, Gopher, WAIS, the World Wide Web, and a few other minor ones like IRC and MUDs. Feel free to flip back here and browse if you're confused about basic usage or what might be important to look for in a client program.

Enough about all the Internet services. But, before we go on and talk about ways you can get Internet access, I should explain about all the different file formats that you run into on the Internet. They're a source of confusion for many new users, so let's move on to chapter 10, "File Formats."

Chapter 10



File Formats

On the Macintosh, we're all used to the simple concept of double-clicking on a document icon to open it in the proper application. The Macintosh keeps track of which documents go with which applications by type and creator codes. Thus, we tend not to think about file formats as much as people who use operating systems that lack the Mac's elegance. Nonetheless, every Mac file does have a format, and if you've ever seen the "Application not found" message, you may have wished for an easier way to determine a file's format.

Note

Various utility programs such as Apple's free ResEdit and PrairieSoft's commercial DiskTop can show you the type and creator codes that the Mac's Finder uses to link documents and applications. You can find ResEdit at:

`ftp://ftp.info.apple.com/Apple.Support.Area/Apple.Software.Updates/Macintosh/
Utilities.Software/ResEdit.2.1.3.sea.hqx`

When you start exploring, you quickly discover that most files on the Internet have filename extensions, as is standard in DOS. Unlike DOS, Unix allows long filenames, so you don't have to think of meaningless eight-character names for everything. Extensions are extremely useful on the Internet because they identify what sort of file you're looking at. On the Mac, you see a different icon or you can double-click on the file and see what program launches, but on the Internet, all you get is the filename and extension.

Unlike standard DOS usage, in which every program seems to have at least one or two extensions for its documents (.wk1, .wks, .doc, .wp, .dbf, .ndx, .idx, and other thoroughly memorable three-letter combinations), a limited set of extensions is commonly

used for files that Mac users care about. These extensions fall into three basic categories: those used to indicate *ASCII* encoding, those used to indicate compression formats, and several others used to mark certain types of text, graphics, sound, and video files.

ASCII Encoding

Programs and other binary data files (files with more than just straight text in them) contain binary codes that most email programs don't understand, because email programs are designed to display only text. Binary data files even include data files, such as word processor files, which contain formatting information or other nonprinting characters. Most programs enable you to save your files in a variety of formats, including text. If you don't explicitly save a file in some kind of text format, then it's probably a binary data file, although there are exceptions.

Computers of different types generally agree on only the first 128 characters in the ASCII character set. (*ASCII* stands for *American Standard Code for Information Interchange*.) The important fact to remember is that after those first 128 characters, which include the letters of the alphabet and numbers and punctuation, a Mac's accented letter may be a DOS machine's smiley face.

Still, people want to transfer files via email and other programs that cannot handle all the possible binary codes in a data file or application. Programmers therefore came up with several different ways of representing 8 bits of binary data in 7 bits of straight text. In other words, these conversion programs can take in a binary file such as the Alarm Clock desk accessory, for instance, and convert it into a long string of numbers, letters, and punctuation marks. Another program can take that string of text and turn it back into a functioning copy of the Alarm Clock desk accessory. I'll leave it to the philosophers to decide whether it is the same program.

Once encoded, that file can travel through almost any email gateway and be displayed in any email program, although it's worthless until you download it to the Mac and decode it. The main drawback to this sort of encoding is that you must always decode the file before you can work with it, although many programs on the Mac decode for you automatically. In addition, because you move from an 8-bit file to a 7-bit file during the encoding process, the encoded file must be larger than the original, sometimes by as much as 35 percent.

Now that you understand why we go through such bother, the Internet uses three main encoding formats (see Table 10.1): *BinHex*, *uuencode*, and *btoa* (read as "b to a"). The only one you're likely to see in normal practice is *BinHex*; to read more about the others, check out the HTML version of *Internet Starter Kit for Macintosh* on the CD.

Table 10.1
ASCII Encoding Formats

<i>Format</i>	<i>Advantages</i>	<i>Disadvantages</i>
BinHex	Macintosh standard	Least efficient
uuencode	Used by LAN email gateways	Doesn't encode resource fork
btoa	Most efficient	Least common

BinHex

BinHex is by far the most common format you see in the Macintosh world because it originated on the Mac and works best with Macintosh files. In fact, it's basically used only on Macintosh computers. You can identify most BinHex files by the **.hqx** extension they carry. I haven't the foggiest idea why it is **.hqx** instead of **.bhx** or something slightly more reasonable. Keep in mind that BinHex is another one of these computer words that works as a verb, too, so people say that they binhexed a file before sending it to you.

Note

*There's another reason for using BinHex when working with Macintosh files. Macintosh files have two forks, a data fork and a resource fork. Most other operating systems don't understand different forks, so you need a way to combine the forks into a single file suitable for uploading to a different machine. There are two basic ways to do this, MacBinary and BinHex. MacBinary creates a binary file that combines the two forks, whereas BinHex combines them in a text file. Since text files have other advantages in terms of surviving mail gateways, people use BinHex far more often. MacBinary files must be downloaded in binary mode and usually end with a **.bin** extension. Most programs that can download files understand MacBinary, but if they don't you can get a stand-alone version at:*

<ftp://ftp.tidbits.com/pub/tidbits/tisk/util/>

There are two flavors of BinHex, but they aren't interchangeable. The BinHex 4.0 format was originally created by Yves Lempereur and has been around forever. BinHex 5.0, which also came from Yves, is more recent but unfortunately causes massive confusion because it doesn't turn binary files into ASCII. Ignore BinHex 5.0 entirely, because everyone else does.

Note

BinHex 4.0 is a file format, and numerous programs can encode and decode that format. Yves wrote a program called BinHex 4.0 years ago, but it has some known bugs and should be avoided. I recommend that you use Aladdin's free StuffIt Expander for debinhexing files, especially because it can also expand various compression formats, and because it's on the CD.

Every BinHex file starts with the phrase **(This file must be converted with BinHex 4.0)** even if another program actually did the creating. Then comes a new line with a colon at the start, followed by many lines of ASCII gibberish. Only the last line can be a different length than the others (each line has a hard return after it), and the last character must be a colon as well (see figure 10.1). Occasionally, something happens to a BinHex file in transit and one line is shortened by a character or two or even deleted. When that happens, the file is toast.



Figure 10.1 Example of BinHex.

BinHex suffers from only two real problems other than a confusing name. It is perhaps the least efficient of the three encoding formats, which means that it wastes more space than other formats. Oh well, just because something is the standard doesn't mean it's the best. Its other real problem is that even though tools exist for debinhexing files on other platforms, they aren't common. Use uuencode if you plan to send binary files that have only a data fork (such as Microsoft Word files, for instance) to a user on another platform.

Compression Formats

Along with the various ASCII encoded formats, on the Internet you frequently see a number of file extensions that indicate the files have been compressed in some way. Almost every file available on the Internet is compressed because disk space is at a premium everywhere.

Unfortunately, because the majority of Macintosh files stored on the Internet are binhexed after being compressed, you don't see the full benefit of the compression. Nevertheless, if your original file is 200K and a compression program reduces it to 75K, you're still on the winning side even if binhexing the file increases it back up to 100K.

The folks who run the Internet file sites like two things to be true about a compression format. First, they want it to be as tight as possible, so as to save the most space. Second, they want to be sure that the files stored in that format will be accessible essentially forever. This requires the format of the compressed files to be made public; in theory, any competent programmer can write a program to expand those files should the company go out of business or otherwise disappear.

This second desire has caused some trouble over the years because the compression market is hotly contested. Companies seldom want to put their proprietary compression algorithms (the rules by which a file is compressed) into the public domain, where their competitors can copy them. For a while there was a project on the Internet to create a public format based on some other public compression formats, but it never saw the light of day. As it is, the only compression format widely available in the Macintosh world that is also public domain is that used by StuffIt 1.5.1, an older and less efficient version of the now-proprietary StuffIt 3.0 format.

Most people on the Internet compress Macintosh files in one of three ways: StuffIt, Compact Pro, or as a self-extracting archive. In addition, there are at least three or four other programs that can compress files, but few people ever use them for files posted on the Internet, other than for self-extracting archives.

StuffIt 3.0

Perhaps the most popular Macintosh compression format on the nets today is StuffIt 3.0, which is used by a family of programs—some free (StuffIt Expander), some shareware (StuffIt Lite), and some commercial (StuffIt Deluxe)—from Aladdin Systems. StuffIt files always have the **.sit** extension. The only confusion here is that the StuffIt file format has gone through three main incarnations: 1.5.1, 2.0, and 3.0. The latest versions of all the StuffIt tools can read all of those formats, but not surprisingly, StuffIt 2.0-class tools can read only files created in 2.0 or 1.5.1, and StuffIt 1.5.1 can read only files in its specific format.

Note

Although the StuffIt file format is version 3.0, the latest version of the programs is 3.5—the file format didn't change when Aladdin revised the programs.

This limitation leads to the common problem on the nets whereby people download a file in StuffIt 3.0 format assuming they can expand it with a StuffIt 2.0-class program because of the **.sit** extension. Unfortunately, because all three file formats use the **.sit** extension, the extension provides no useful indication, and StuffIt 2.0 spits up all over a StuffIt 3.0 file. The simple solution to this problem (and most other compression problems) is—and listen carefully—StuffIt Expander.

StuffIt Expander can expand any StuffIt format, it can expand Compact Pro archives, and as an added bonus it can debinhex files as well. It slices, it dices, and... let me just say that no one should be without StuffIt Expander. That's why I put it on the ISKM disk. Thanks are due to Aladdin Systems for making such a useful tool available for free. You can find the latest versions of StuffIt Expander and StuffIt Lite online in the directory:

<ftp://ftp.tidbits.com/pub/tidbits/tisk/util/>

There's actually a caveat to StuffIt Expander being free. Aladdin Systems has released a shareware program called DropStuff with Expander Enhancer which, when you pay your shareware fee, gives StuffIt Expander the capability to expand basically all the file formats that the full StuffIt Deluxe can expand through its translators. In addition, you can stuff files quickly by merely dropping them on DropStuff, so it's a doubly useful program. I strongly recommend you register DropStuff with Expander Enhancer if you intend to work with many different file types from the Internet. Another way to get the same functionality is to buy one of Aladdin's commercial programs, such as StuffIt Deluxe or SITcomm.

Compact Pro

Compact Pro, a shareware compression utility from Bill Goodman, is almost as popular as the StuffIt family in the Macintosh world. Functionally, StuffIt and Compact Pro do the same thing—create a compressed archive of one or more files. In my experience, both utilities do an admirable job, so personal preference and other features may sway you one way or the other.

Compact Pro files are always identified by their **.cpt** extension. You may see an earlier version of Compact Pro floating around on the nets as well. It's called Compactor and uses the same file format as Compact Pro, so you don't have to worry about which version created a given file. Compactor is just an older version of Compact Pro, but Bill Goodman had to change the name for legal reasons. You can find Compact Pro in:

<ftp://ftp.tidbits.com/pub/tidbits/tisk/util/>

Self-Extracting Archives

What if you want to send a compressed file or files to a friend who you know has no compression utilities at all? Then you use a *self-extracting archive*, which is hard to describe further than the name already does. Most compression programs on the market can create self-extracting archives by compressing the file and then attaching a *stub*, or small expansion program, to the compressed file. The self-extracting archive looks like an application to the user, and if you double-click on a self-extracting archive, it launches and expands the file contained within it. Internet file sites prefer not to have many files, particularly small ones, compressed in self-extracting archives because the stubs are a waste of space for most people on the nets, who already have utilities to expand compressed files.

You can always identify self-extracting archives by the **.sea** extension. You can tell by the icon which compression program created any given self-extracting archive, but on the whole it makes no difference.

Note

You may want to keep in mind a number of other file type issues. These issues, relating to different formats of text, graphic, sound, and movie formats aren't generally a problem though, so if you want to learn more about them, check out the HTML version of Internet Starter Kit for Macintosh on the CD.

Format's Last Theorem

I think we've had plenty of background material on the main file formats that you see on the Internet, so let's move on and look at the software that provides your connection to the Internet, MacTCP and MacPPP.

Part III



Full Internet Access

We've finally hit the really good stuff! I titled this part of the book "Full Internet Access" because everything else we've talked about so far has been background material. With full Internet access, as discussed in this section, you will find that your only limitations are how fast your modem can go and how much time you want to spend using the Internet.

Chapter 11, Making Connections

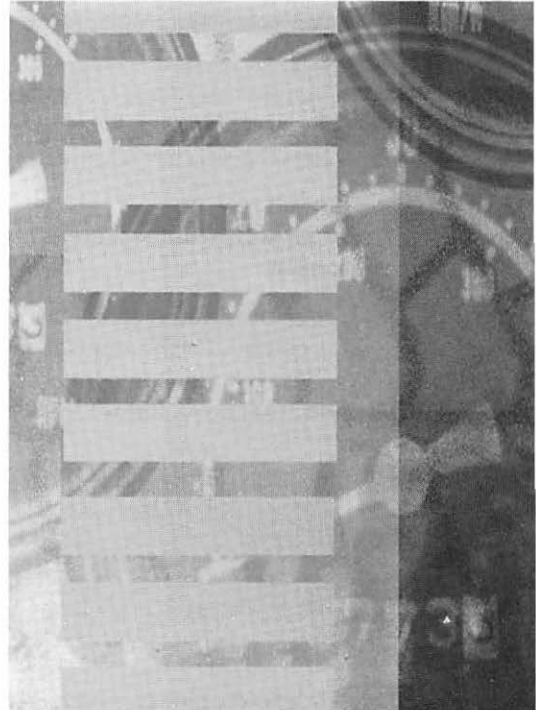
MacTCP is the most necessary piece of the connection puzzle, because without MacTCP, none of the MacTCP-based programs will work well or at all. Although it requires some configuring, MacTCP is really quite simple. The ISKM Installer prevents you from needing to do much at all.

MacTCP by itself won't make your Internet connection, though, and for that you need MacPPP. In the "MacPPP" section, I go through the steps of configuring MacPPP, the freeware version of PPP installed by the ISKM Installer, and then discuss some of the more common problems (and their solutions) that you may encounter.

Chapter 12, MacTCP-based Programs

On to the MacTCP-based programs that you use to do things on the Internet. Although chapter 11 will get you on the Internet with a MacTCP connection, nothing happens until you run a MacTCP-based program. That's what I cover in chapter 12, starting with "Email" and "Usenet News," and continuing on to "FTP" and "World Wide Web." In the "Email" section, I discuss the free Eudora, the best Internet email program available. The "Usenet News" section concentrates on another excellent program, John Norstad's free NewsWatcher. For "FTP," I look at Anarchie, a \$10 shareware program from Peter Lewis, which may be the best FTP program for any type of computer. And finally, in the "World Wide Web" section, I discuss two popular Web browsers, MacWeb and Netscape Navigator, along with the various helper applications that you use for types of information like sounds and movies that the Web browsers don't understand.

Chapter 11



Making Connections

"The time has come," the Walrus said,
"To talk of many things:
Of news—and chips—and Gopher hacks,
Of Babbage's—and pings."

Apologies to Lewis Carroll, but the time has come to talk of many things, all of them dependent on Apple's MacTCP. I'm going to start by discussing MacTCP itself, which Hayden licensed from Apple to include on the *Internet Interactive* CD-ROM.

MacTCP

Roughly speaking, MacTCP is a translator. It enables the Macintosh to speak the language of the Internet, TCP/IP (*Transmission Control Protocol/Internet Protocol*). Normally, of course, Macs speak AppleTalk to one another, over Macintosh networks. You must have the MacTCP control panel installed and configured properly in order for the MacTCP-based programs, such as Eudora and Anarchie, to work, although MacTCP does not make the connection itself. Think of MacTCP as the Babel Fish from the *Hitchhiker's Guide to the Galaxy*. Pop it in your Mac's ear (the Control Panels folder, actually), and your Mac understands the Internet noise that flows in and out.

The metaphor of speaking and languages isn't quite accurate because TCP/IP is actually a transport protocol. But the idea of MacTCP as a Babel Fish that translates Internet gibberish into a language the Mac can understand seems to be the most understandable metaphor. Luckily, everything that MacTCP does happens at such a low level that you

never notice. In fact, after you set up MacTCP correctly, you should never notice that it's present.

Once your Mac is connected to the Internet with MacTCP via a local area network or PPP, it is essentially the same as any other Internet machine and has its own IP number. This means that you can connect to other Internet machines directly, without going through an intermediate machine.

Note

Because the Internet is based on the TCP/IP protocols, the only way for a Mac to enjoy a full Internet connection is to use MacTCP. If you do not have MacTCP installed and a MacTCP-based connection using PPP or an Internet-connected network, you cannot use the MacTCP-based programs. Period.

Apple and other companies have thought in the past that MacTCP is a program that only large organizations want to buy, install, and configure. It's a poor assumption these days, because individuals using PPP can easily gain access to the Internet, and PPP requires MacTCP. However, if you work at a university or business that provides your Internet connection, it's a good bet that you have a network administrator who knows a great deal about MacTCP, and who has probably preconfigured it for your convenience.

Note

Evidence that this view of MacTCP is changing comes from Apple's inclusion of MacTCP in System 7.5. If you have System 7.5, even if you've never tried to connect to the Internet, you very well may have MacTCP already installed (but not configured).

The ISKM Installer on the CD installs MacTCP and configures it for use with EarthLink Network; the Leonardo ISKM Installer installs MacTCP and configures it for use with Leonardo Internet (you don't need to worry about this second option unless you live in the Los Angeles area). It also assumes you want to use MacPPP, so if you wish to use the software included on the CD with a network connected to the Internet, you should consult your network administrator to see what you should and should not install (MacPPP, for instance, is unnecessary if your network is connected to the Internet).

On the CD

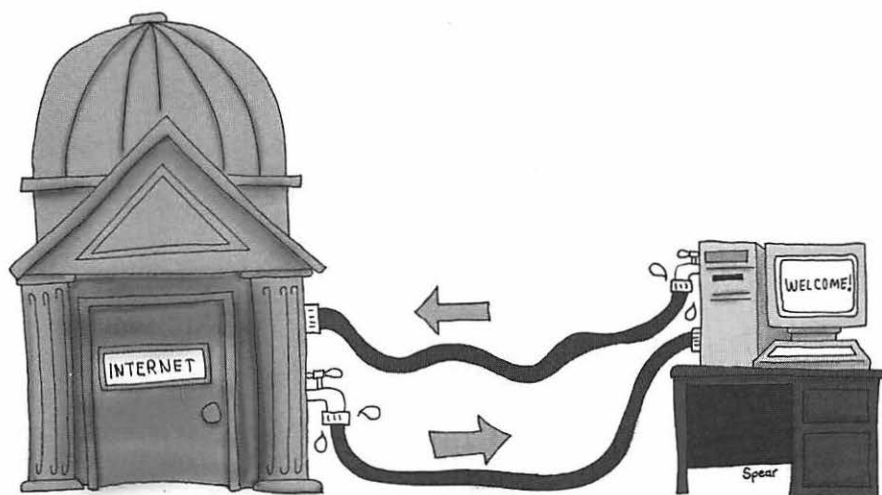
For more information about MacTCP, how to configure it manually, and a number of utilities you can use with MacTCP, and for a look into the future at Apple's forthcoming MacTCP replacement, Open Transport, read chapter 17, "MacTCP," in the CD version of Internet Starter Kit for Macintosh.

MacPPP

An ever-increasing number of people who don't work at a large business or university want access to the Internet, and an equally ever-increasing number of Internet access providers are springing up to meet that need. Because individuals seldom have the level of connectivity enjoyed by those in business or education, they must make do with slower connections. Until recently, they've had to cope with clumsy Unix shell accounts as well. That clumsiness is avoidable now that PPP (Point to Point Protocol) accounts have become widely available; more importantly, they provide access to some extremely cool software that I talk about in the next chapter.

PPP provides an Internet connection for people who connect via modem. If you have a network, either Ethernet or LocalTalk, connected to the Internet, you don't need PPP.

The easiest way to understand PPP is to pretend that you don't have water service inside your house. Every time you want to take a shower, you must run a garden hose out to the water hookup outside, take your shower, and then reel the hose back in. That's exactly what PPP does—it establishes a temporary, low-speed connection to the Internet. You must create that connection before you can run programs such as Anarchie and Netscape.



Note

Although MacPPP allows you to launch a MacTCP-based program without connecting first (the PPP software sees what's up and then establishes the connection automatically), I've found that auto connect features can be flaky. If you can use it, great, but if it doesn't work reliably, connect manually first.

The free MacPPP, written by Larry Blunk of Merit Network, is my dialup connection method of choice. This is primarily because of its simple configuration and setup, and partly because it works well for almost everyone. We've included it on the CD and the ISKM Installer installs it properly for you.

Installation and Setup

The ISKM Installer properly places a control panel called Config PPP in your Control Panels folder, and an extension called PPP in your Extensions folder. It also ensures that MacTCP is properly configured to use PPP.

Open the Config PPP control panel. As you can see in figure 11.1, Config PPP has a vaguely clunky interface, but it is almost effortless to configure.



Figure 11.1 Config PPP control panel.

The Port Name pop-up menu enables you to choose the Modem port or Printer port (or any other ports registered with the Communications Toolbox); it is usually set to Modem port.

Note

If you use a PowerBook with an internal modem, the Port Name pop-up menu may confuse you. A few PowerBook modems, including the Apple Express Modem, the Global Village PowerPort/Mercury for the PowerBook 500-series, and Duos, are bus modems, and show up in the menu as Internal Modem. Other internal PowerBook modems are non-bus modems and use an internal connection to the Modem port. For these, choose Modem port. Mac AV users may also have a GeoPort option.

The Idle Timeout pop-up enables you to set a time of inactivity, from five to 120 minutes, after which MacPPP will close the connection. If you do anything during this time, the timer resets and MacPPP starts counting again. If your connection remains idle for the duration specified, MacPPP closes the connection. If you have the Quiet Mode checkbox checked, MacPPP does so without warning; if not, then at the end of the idle time period MacPPP presents you with a dialog that enables you to either ignore the warning and leave the PPP connection active, or close PPP.

I haven't used this feature much, because I simply connect when I want something and disconnect when I'm done. However, I see two tremendous uses for this feature. First, for those people who pay by the minute for their connections, having MacPPP hang up if the line is idle could save you a fair amount of money. Second, if you want to download a large file before you go to bed, simply set MacPPP to a relatively short timeout value and it will hang up when it's done downloading the file. Make sure Quiet Mode is checked if you want it to hang up without confirmation.

Note

Some people have experienced problems with MacPPP connecting seemingly randomly. Although most cases are caused by a program like Anarchie or TurboGopher asking for MacTCP services, it seems that on occasion MacTCP simply decides to do something and asks MacPPP to dial out. If you experience this, setting a short idle time in MacPPP keeps those unwanted connections as short as possible. And of course, if you shut off your modem, MacPPP won't be able to dial out at all.

It appears that MacPPP does a *soft close* in this idle timeout situation, which means that a MacTCP-based application can automatically re-open the connection by requesting MacTCP services. If MacPPP did a *hard close*, applications wouldn't be able to re-open the connection automatically; you would have to click the Open button to open a new connection.

Note

I can't predict how different MacTCP-based applications will behave if their connection disappears due to the line being idle for five or ten minutes. If you anticipate being in a situation where MacPPP might hang up automatically after an idle timeout, make sure to save your work in all other open applications. Some applications may even hang your Mac, so be careful.

The Echo Interval pop-up menu provides the opportunity to configure MacPPP to periodically query the line to see if your connection has dropped. If MacPPP receives no response after three successive requests, MacPPP assumes that the connection has gone dead. I've always left Echo Interval turned off, although if you have trouble with your connection dropping frequently, using it may make life easier. When MacPPP detects a

dead connection, it pops up a dialog box with three buttons for Close PPP, Ignore, or Restart, which in this case means restarting the PPP connection, not the Macintosh. Of course, if your connection drops, you must at least quit open MacTCP applications before trying to do anything else. You may have to restart to clear things up appropriately if the open applications have become sufficiently confused by the loss of the connection.

Note

The Echo Interval feature continually sends packets to the server and waits for a response, but these packets don't count as traffic for the Idle Timeout feature.

The Terminal Window checkbox is one of MacPPP's most useful features. If you check it, MacPPP ignores the Phone number and Modem init fields in the Configure Server and Connect Script dialog boxes (I'll get to them in a minute). Instead, it makes you walk through the connection manually, starting with dialing the modem with an ATDT command.

Note

In some situations, MacPPP's terminal window doesn't echo what you type back to the screen, but the characters will be sent when you press Return.

You may never need to use MacPPP's terminal emulator, but if you have trouble logging on, it's much easier to have the terminal emulator built into MacPPP rather than be forced to use an external one.

If you must use the terminal window, dial the modem manually, enter your username and password, and once you start seeing some gibberish characters that indicate the start of PPP data, click OK to start the PPP session.

Note

I strongly recommend that you use MacPPP's terminal window if you're having trouble getting connected because it often reveals problems on the provider's end (like the provider didn't set up a PPP account for you) and eliminates any mistakes you might have made in the connect script.

The two final checkboxes in the window are Hangup on Close and Quiet Mode. Hangup on Close, if checked, sends the Hayes +++ escape string and then an ATH to hang up the modem. My modem hangs up fine without this checkbox selected, but if yours doesn't, select it. The Quiet Mode checkbox prevents certain actions from requiring confirmation, most notably the Idle Timeout disconnect.

Once you've configured the main part of the Config PPP control panel, you should be able to select, from the Server pop-up menu, one of the configurations that's named for your modem. If you don't have one named for your modem, pick one that sounds similar. If you wanted to create an entirely new configuration and figure out the modem initialization string for yourself, you could click the New button, name it, and then continue. I don't recommend doing that since I've done some of the necessary configuration for you. Click the Config button to bring up the Configure Server dialog box (see figure 11.2).

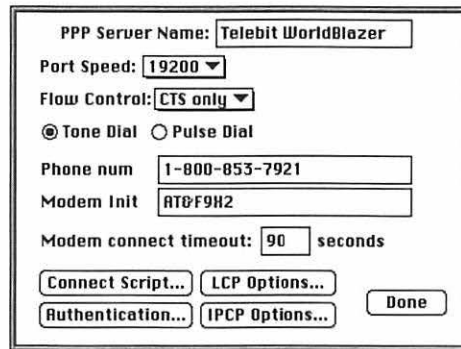


Figure 11.2 MacPPP Configure Server dialog.

If you want to rename your server configuration, you can edit it in the PPP Server Name field. It makes absolutely no difference what you set the server name to—it's only for your use. In the version of the PPP Preferences the ISKM Installer puts on your disk, I use the PPP Server Name field to identify different modems, for instance.

After you set the PPP Server Name, you can configure the basic modem variables. Set the Port Speed pop-up menu as high as it will work with your modem (my WorldBlazer doesn't like speeds over 19,200 bps by default), and always try to use a Port Speed faster than your modem, so it can take advantage of the modem's compression capabilities.

The Port Speed is the speed at which the Mac and the modem communicate, **not** the speed at which the two modems communicate (unless it happens to be slower than the fastest speed the modems have in common, at which point it forces the modems to communicate at that speed). Do not set the Port Speed menu to 14400 or 28800 even if you have a 14,400 bps or 28,800 bps modem. The reason is that some modems don't accept those as valid port speeds, and MacPPP won't talk to the modem properly.

Note

Port speed is reportedly irrelevant if you use a bus modem, since it doesn't use the modem port. I'd set it to 57600 if you have one of these modems.

Modem flow control, also sometimes called handshaking, is one of the most confusing topics in telecommunications on the Macintosh. The basic idea is that the flow of the incoming and outgoing packets must be organized and controlled or else you experience the packet equivalent of traffic jams and accidents. For a modem to do handshaking in hardware, you need a special cable (luckily, one that comes with most fast modems sold for Macs, although every now and then someone has trouble because you can't use high speeds on modern modems without one). All of the various options we have discussed are forms of hardware flow control or hardware handshaking. PPP can, in theory, use software flow control, also called XON/XOFF, which doesn't require a special cable, but I've never seen anyone try. Usually, there's no point.

The Flow Control pop-up menu has options for None, CTS only, RTS (DTR) only, and CTS & RTS (DTR). I recommend that you try setting CTS & RTS (DTR) as the first try. The problem with the CTS & RTS (DTR) setting is that you may experience random hangups when lots of data is coming in unless your modem init string includes &D0, or, if your init string uses &D2, it is also properly configured to ignore short periods of DTR going low (check your modem manual for the S25 S-register setting and try setting it to 50 in the modem init string). If that's true, try dropping down to CTS only. I ran into some of the hangups, and haven't had any trouble at all since I switched to CTS only.

Note

Internal PowerBook bus modems can have the Flow Control pop-up menu set to None; all other modems must use some sort of flow control.

Tone Dial versus Pulse Dial should be obvious based on your telephone. Tone dialing is far more prevalent these days. The phone number field should be self-evident, although you may have to add special prefixes to get through a company or hotel phone system. Adding the prefix of either 8 or 9 and a comma is a common solution (the comma ensures a short pause after sending the 8 or the 9 and before dialing the rest of the phone number).

Note

If you must enter a lot of digits in the phone number field to account for a calling card number, you may have to eliminate all the dashes, which aren't necessary and take up space in the field. I forget the maximum number of digits it can take, but I ran into it once when using a calling card number.

Next we come to the Modem Init field. Modem initialization strings have been an unending source of headaches for Internet users using modems. My recommendation is to start with the factory default configuration for your modem (usually AT&F, AT&F1, or AT&F2, although the numbers change depending on the modem).

Note

If you use the ISKM Installer to install MacPPP, you get a PPP Preferences file that contains a number of likely (but untested) modem strings for many modem types. In addition, even more modem strings are located in a file called Modem Strings in the MacPPP 2.0.1 folder in your ISKM3 Folder.

I should make one uncommon but important point. Make sure that XON/XOFF, or software flow control, is turned off. On at least one modem, the Telebit QBlazer, MacPPP fails badly if XON/XOFF is active, and the MacPPP documentation notes that software flow control should be turned off as well, so I'm assuming that XON/XOFF is Public Enemy Number One with PPP accounts.

Note

You may wonder what the X2 is doing in my initialization string. I discovered that my Telebit WorldBlazer doesn't return the BUSY code in the default X1 setting, and if the modem doesn't return that BUSY code, MacPPP has no way to know that the line is busy and it should redial (which it does automatically). Other modems may have a similar problem.

Finally, the Modem connect timeout field offers you a chance to increase the amount of time MacPPP will wait for the connection to occur. If it takes MacPPP a long time to negotiate your connection, you may need to increase this value.

As more alert readers will have noticed, I haven't yet mentioned where you enter your userid and password. There are two possibilities here, depending on what the PPP server on your host machine supports. If you're lucky, you can use PPP's Password Authentication Protocol (PAP) to negotiate your connection. If your server doesn't support that, you instead must use a connect script. First, the Authentication dialog (see figure 11.3).

As you can see, all you do here is enter your Authentication ID and Password into the appropriate fields. The Auth. ID field holds your userid and the Password field holds your PPP password, appropriately hidden so others can't see it. If you're really paranoid about your password, you can leave that field blank and MacPPP will prompt you to enter it every time you connect.

Note

If you work with EarthLink Network using any of their TotalAccess services (Southern California, 800, USA), you can use Authentication. However, a simple connect script would also work, and with providers other than EarthLink, may be required instead of Authentication.

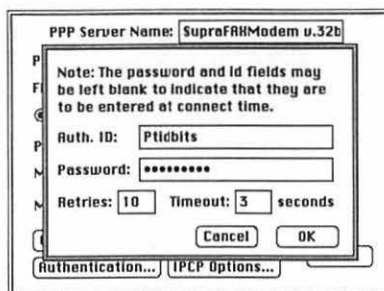


Figure 11.3 MacPPP Authentication dialog.

Click OK to save your changes and return to the Configure Server dialog. If you use the Authentication dialog, you need not enter anything in the Connect Script dialog, and in fact, you're best off deleting anything that might be in there. If the Connect Script dialog is empty, MacPPP sends the authentication information without worrying about looking for the login and password prompts, which might screw up the entire process.

Note

If you leave the Connect Script blank or configure it incorrectly, and leave the Authentication dialog empty, MacPPP prompts you for your userid and password, just as you would have entered them in the Authentication dialog.

If using the Authentication method of logging in to your PPP server machine doesn't work, you must instead use the Connect Script dialog to script your way in. It's still pretty easy.

Basically, all you do in the Connect Script dialog is replicate the process of logging in to the host manually (and the terminal window can be helpful in figuring out the connect script). You click an Out button to indicate that MacPPP should send the contents of the field to the left (and a carriage return if the checkbox is selected for that line), and you click a Wait button to indicate that MacPPP should wait for the string specified in the field to appear before moving on to the next line.

Note

If you're unlucky, connecting to your PPP server will require more than the eight fields MacPPP provides here. In that case, if your server doesn't support authenticated logins as discussed previously (which is likely, if it requires more than four send/expect interactions), your only option may be to use the commercial InterPPP. Actually, that's not entirely true, since you can do some funky scripting within MacPPP by using special codes like \t (which drops you into the terminal window at that point in the script), \r (which sends a Return to the host), and \d (which forces a short delay). The trick when creating such a hacked script is to use delays instead of the Wait buttons, thus enabling you to cram more functional script lines into the same space.

Meanwhile, back in the Configure Server dialog, you've probably been wondering what's inside the LCP Options dialog and the IPCP Options dialog. You really don't want to know, and I'm not going to show you. Suffice it to say that I'll be very surprised if you know what to do with the options in there. I certainly don't, and the MacPPP documentation says that no normal user should ever have to change any of those settings.

Well, that's it for configuring MacPPP. It's quite easy, and in fact, it may be the easiest method of connecting to the Internet.

Basic Usage

Once it's configured, using MacPPP is a piece of cake. It works in both manual and automatic modes, so you can either click the Open button in Config PPP or you can merely open a MacTCP-based program that opens MacTCP, at which point MacPPP connects automatically. I've found MacPPP's auto-connect feature to be fairly reliable, although not absolutely guaranteed. It seems that some applications don't quite play by the rules, and those applications won't work in auto-connect mode.

Note *MacPPP's auto-connect feature only works on the second or subsequent connections after you restart your Mac if you use Soft Close to close your connection. If you use Hard Close, MacPPP cannot auto-connect. Nonetheless, there are some cases of MacPPP dialing for apparently no reason that are eliminated with Hard Close, so that's what I always use.*

Once you click the Open button or have MacPPP connect automatically, you see some dialogs indicating that MacPPP is dialing, logging in, establishing the PPP link, and checking for the network. After those dialogs, the little faces under the Open button will become happy faces and the connection will be complete.

Note *Be careful of the auto-connect mode—if you do something like put Netscape in your Startup Items folder, it will launch on every startup and make MacPPP establish a connection to the Internet. You're unlikely to do that with Netscape, but certain control panels or extensions also can activate the auto-connect feature, which can be a pain.*

Once you've established a PPP connection with MacPPP, you can run any of the MacTCP-based applications and do whatever you want.

When you're done with your work and want to close your connection, first quit all of your MacTCP-based applications. Some of them dislike having the connection disappear from under their little electronic feet. To pull the plug on the PPP connection, you can do one of two things, depending on how you use MacPPP. First, you can click the Hard Close button, which hangs up the connection and "locks" MacPPP so that the only way to establish a new connection is to click the Open button. This prevents any applications

from forcing MacPPP to open the connection automatically while, say, you're not present. Second, if it doesn't bother you to possibly have applications dialing your phone behind your back, you can click the Soft Close button to close a connection. That leaves open the auto-connect feature for the rest of that session, so launching an application makes MacPPP establish a new connection.

MacPPP Details

Larry Blunk and Merit Network deserve kudos for making such a fine program available to the Macintosh Internet community for free. However, I should note that they cannot support MacPPP, so if you have problems or questions about it, you should ask on a mailing list like the Apple Internet Users mailing list (send a message containing the command **info apple-internet-users** to **listproc@abs.apple.com** for information on how to subscribe) or in a newsgroup like **comp.sys.mac.comm**. You can retrieve the latest version of MacPPP on the Internet from either of the following:

```
ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/
```

```
ftp://merit.edu/pub/ppp/mac/
```

Note

For more information about utility programs you can use to make working with MacPPP easier, and for a look at a couple of other PPP programs for the Mac, scan to the end of chapter 18, "PPP," in the CD version of Internet Starter Kit for Macintosh.

But What If...

The silly part of what I've written above is that once you have installed and configured MacTCP and MacPPP, your only interaction with them is opening and closing the connection. However, that assumes everything goes as planned, and as we all know, Murphy's Law has yet to be declared unconstitutional. Of course, if everything is working properly with your connection, in defiance of Murphy and the laws of nature, feel free to ignore the next section entirely.

Troubleshooting Your Connection

If you have installed and configured MacTCP and MacPPP and everything works, congratulations! Skip this section entirely; it can get a bit technical in places and if everything works, there's no need to dwell on what might not work. Most people don't have much trouble, but there are some pitfalls to avoid and some tricks and tips I've learned. I'd like to share some of the problems and solutions with you here, and although I hope you

don't need to read this section, if you do, I hope it helps. I've formatted the section along the lines of an Internet FAQ, or Frequently Asked Questions, list. Within each section, I've tried to organize the question and answers roughly as you might experience them—for example, there's no point in putting suggestions about improving performance before suggestions about how to get around a modem problem and connect in the first place.

Reporting Problems

Before I even begin to talk about what might go wrong, I want to say a few words about how you can best go about isolating problems and then reporting them on the nets or to tech support. If you ask for help on **comp.sys.mac.comm** by posting a note that says something like, "I'm connecting to the nets via PPP and it doesn't work. What am I doing wrong?" you won't get any helpful responses. You probably won't get any responses at all, helpful or not, since people will have no clue what your problem is other than the fact that you don't know how to ask for help. If you follow the steps that I suggest when working through **any** problem, not only problems with MacTCP or MacPPP, you'll be better off.

When you've determined that you have a problem, do the following: Start over completely from scratch, removing from the System Folder all files related to MacTCP and MacPPP, including preferences files.

On the
CD

One quick way to find all these files for quick removal is to use the Find command in the Finder's File menu to search for "TCP" and "PPP." I recommend not actually deleting these files until you're up and running again, just in case, but that's why Apple created a persistent Trash.

After you've removed the old files, carefully follow each step in the instructions, noting anything that doesn't seem to mesh between your setup and what the instructions say. If you deviate from the instructions, note that, too. In many cases, following this procedure will either solve the problem or reveal where it lies. Taking something apart and putting it back together often fixes problems for no apparent reason. Don't think of this process as an unpleasant chore, because then you're likely to become careless and miss an important clue. Troubleshooting can be a lot of fun, since you learn a lot more about the topic at hand, and you get to solve a real-life mystery in which no one dies.

Unfortunately, because we're all amateur sleuths, we're not always able to find the solution to a problem and must consult others who are more knowledgeable or who have a different way of looking at the problem. If you are having trouble with a commercial program, the first experts to turn to should be the technical support staff at the company that produced your program. I've heard good things about most of the technical support staffs of companies that make Internet applications, although quality tech support is never guaranteed.

When dealing with telephone tech support people, keep in mind that they probably know a lot more about the program in question than you do, they answer a huge number of calls every day, and the job has a high burnout rate because it's so stressful. You're most likely to get the best help if you're polite and cooperate with what they ask you to do. If you call and announce that you're a power user and why doesn't this stupid program work anyway, you're unlikely to get decent help. If, on the other hand, you call, say that you're having troubles, and give the information the tech support person asks for, she can do a much better job. It never pays to alienate the person whom you're asking for help—whatever is wrong is almost certainly not her fault.

If you are using a freeware or shareware application, it usually says whether or not the author is willing to help via direct email. One way or another, though, there are several places where you can ask for help from other users, many of whom are true experts. Also, the developers of many of the freeware and shareware utilities tend to hang out in these same places and help their users, even if they prefer not to be continually slammed by personal email. The best place to ask for help with Internet stuff is on the Usenet newsgroup **comp.sys.mac.comm**. There also are many knowledgeable people who hang out in the Apple Internet Users mailing list, which you can subscribe to by sending email to **listproc@abs.apple.com** with the command **subscribe apple-internet-users Your Name** in the body of the message. If you can post to a newsgroup, there are often newsgroups specific to your provider where local folks hang out and answer questions.

Note

If you don't read a newsgroup regularly, you should explicitly ask for replies via email so you don't miss any replies.

No matter what, if you want any of these people to help you, you must help them first by sending a complete report. In that report, you should include the following:

- List exactly what it is that you're trying to accomplish. This should be specific, since telling someone that you want to read Usenet news with NewsWatcher isn't salient to the problem of not being able to get MacPPP to connect. Take each goal a step at a time.
- Mention the fact that you have carefully followed the directions. This fact tells people that (a) you're not a complete idiot and can read, (b) that you have gone through a certain set of procedures already, and (c) it tells them that you are capable of following any suggestions they make. If you haven't followed the directions carefully, don't bother posting until you do so.
- List the salient facts of your software and hardware setup. Include things such as what Mac you have, what modem you use, what version of the System software you're running, and any weird stuff that you can't eliminate from the testing (such as, for example, an Outbound user who had additional software from Outbound that was necessary for his machine to boot properly). Unless asked, don't bother

listing out every extension and control panel on your hard disk. You should have already eliminated them in the process of testing by removing everything but MacTCP, PPP, and Config PPP from your Extensions and Control Panels folders. Also, an unusually long report will turn many people off.

- Talk a bit about what you have already tried, whether or not it worked, and whether you noticed anything strange happening at any time during the process. If you encounter error dialogs at any time, report exactly what they say.
- Be nice. The last thing you want to do is insult an expert's favorite program, since they're less likely to help you at that point. When you're in trouble, it doesn't help to alienate anyone.

MacTCP Q & A

Before anything else, let me emphasize that you may need to reinstall a clean copy of MacTCP at various times to solve problems. Thus, you must keep a clean copy that you have never opened on a locked floppy. The copy that comes on the CD qualifies as a clean copy of MacTCP.

Note

At some point you may want to set aside a clean, unused copy of MacTCP to facilitate reinstallations. To do so, move your existing copies of MacTCP and MacTCP Prep to the desktop, restart, run the ISKM Installer and install just a new copy of MacTCP. When forced to restart, keep the extensions off by holding down the Shift key as the Mac boots. This ensures that you have a clean copy of MacTCP in your Control Panels folder. Put that copy of MacTCP on a floppy as a backup. Then replace your old MacTCP and MacTCP Prep files and restart to return to your previous state.

Also, let me recommend that if you have FTP access, you get a copy of Peter Lewis's free MacTCP Watcher. It includes another document on troubleshooting MacTCP connections, written by Eric Behr. It's in:

<ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/>

Anyway, on to the questions and answers about MacTCP!

Q: I don't see an icon for PPP in my MacTCP control panel.

A: Install PPP in your Extensions folder, and try again.

Q: I get a weird -23004 error from MacTCP, and it complains about its drivers not being installed.

A: Make sure that you select PPP and *not* LocalTalk in the MacTCP control panel.

Q: What about that weird slider bar in the upper right of the MacTCP configuration dialog?

A: Ignore it unless you're on a subnet, which means you'll have a network administrator who can tell you what to do there.

Q: Should I type anything in the IP number box in the MacTCP control panel if I'm using Server addressing?

A: No. Only enter a number there if you use a Manually addressed account.

Q: What are the MacTCP DNR and MacTCP Prep files?

A: MacTCP creates them when you restart to store various settings and preferences. You can throw them out at any time with impunity, since MacTCP recreates them with the same settings when you restart. Note that you must restart after throwing them out since MacTCP applications require MacTCP DNR to work properly.

If you reinstall MacTCP without throwing out these files, MacTCP retains the settings it had before you reinstalled. This can be useful for moving copies of MacTCP around, but these files also tend to retain any corruption.

Q: My Mac crashed the first time I restarted after reinstalling MacTCP. Should I be worried, even though it doesn't crash now?

A: I would completely reinstall MacTCP (throwing out MacTCP DNR and MacTCP Prep as well) and reconfigure to be safe, but I don't think you should worry too much. Anti-virus software like SAM can sometimes gets persnickety about MacTCP creating the MacTCP DNR file, and I could see that perhaps causing the problem.

In fact, I hear that if the MacTCP DNR files becomes corrupted, some anti-virus software, such as Gatekeeper, could possibly prevent MacTCP from updating it, which is a bad thing. If you use sensitive anti-virus software that tries to prevent unknown actions (Disinfectant is fine), to be very sure it isn't causing problems, turn it off, delete MacTCP, MacTCP DNR, and MacTCP Prep, reinstall a clean copy of MacTCP, restart, turn the anti-virus software back on, and restart again.

Q: Hmm, I don't seem to have a MacTCP DNR file. Why not?

A: That's really weird, but one reader reported a reproducible conflict with a control panel from Apple called CPU Energy Saver. Try removing it and restarting. MacTCP should create a new MacTCP DNR file on restart if it's not present.

Q: I also don't have a MacTCP Prep file.

A: First, try opening MacTCP, changing something and then closing it. That should force the creation of a MacTCP Prep file in the Preferences folder. If that doesn't work, restart without extensions, particularly anti-virus programs, and try reconfiguring MacTCP again.

- Q: I crashed while using Netscape or some other MacTCP-based program. Should I reinstall MacTCP?
- A: Possibly. First, connect again to see if Netscape works. If it does, you're fine. If it doesn't, throw out the MacTCP DNR file and restart. Try Netscape again. If it still doesn't work, completely reinstall MacTCP from scratch. This isn't usually necessary, luckily. After a truly nasty crash, you may find it necessary to reinstall MacPPP as well.
- Q: I'm getting the impression that reinstalling MacTCP is a common occurrence. Is that true?
- A: Yes and no. I seldom do it, but frankly, if anything goes wrong, reinstalling MacTCP is worth trying. Make sure to throw out MacTCP DNR and MacTCP Prep, too, since they can harbor the corruption that caused MacTCP to have problems in the first place. Always keep a copy of MacTCP on a locked floppy disk, to facilitate reinstalling.
- Q: I'm running MacTCP 2.0.2 or 2.0.4. Should I update to 2.0.6?
- A: Sure, why not? I personally haven't seen any problems in 2.0.2 or 2.0.4 that were fixed in 2.0.6, but others have, and it's a free update that's stored on:

`ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/mactcp-206-updt.hqx`

Besides, MacTCP 2.0.6 is on the CD that comes with this book, so you can install it from there if you want, first removing the old MacTCP, MacTCP DNR, and MacTCP Prep.

- Q: When I went to update my copy of MacTCP to 2.0.6, I got some sort of error about a DRVR 22. What's that all about?
- A: The MacTCP updater only works on a clean copy of MacTCP that has never been opened before. Get a new copy from your master disk, update it, and the updater will work fine. Then, keep a clean copy of 2.0.6 on a locked floppy somewhere for use when reinstalling.

Domain Name Server Errors

I've noticed many people running into a problem where MacPPP connects properly, but trouble arises with the domain name server when they attempt to run any of the other MacTCP programs. Sometimes they crash or hang, but the behavior is usually completely reproducible (although we have seen the occasional exception, where it will work fine once after reinstalling MacTCP but fail on subsequent connections).

I haven't absolutely solved this problem yet, I'm sorry to say, but here are some things to try. I find that the easiest way to test this situation is with Peter Lewis's free MacTCP Watcher, which shows an IP number but no Mac name in this instance. It also complains

about a “No answer error” or a “Cache fault error” when it is unable to find the domain name server. If you use Fetch, you can tell quickly that you are seeing the problem if when you connect, the dog cursor is frozen. If the dog cursor runs, you’re generally better off. If the cursor is frozen, immediately hit ⌘-period to try canceling the connection before Fetch hangs. NCSA Telnet can prove useful as well, since in some cases you may be able to telnet to a machine by using its IP number, but using the domain name fails. This indicates that the connection works, but that there is a problem resolving domain names.

Note that the problem is generally not with your account. Each time someone has had this problem and asked me to check their account, it has worked fine if they did indeed have a PPP account, although trying to connect via PPP to a shell account can cause the problem. That said, here are some things to try, in this order:

- Disconnect, reboot, and reconnect. Sometimes that’s all it takes.
- Throw out your MacTCP DNR file and restart. Sometimes this file becomes corrupted, and MacTCP will create a new one on restart if necessary.
- Check your domain name server configuration in MacTCP carefully to make sure you typed the correct IP numbers. I’ve made this mistake before.
- If you are using a Manually addressed account, make sure you have typed the correct number into the Gateway address box in MacTCP’s configuration dialog. Only your provider can tell you what your Gateway address is—you cannot guess it.
- Login with a terminal emulator just to make sure your account is set up for PPP. If after providing your user name and password and sending any necessary commands to start PPP (ask your provider if any are necessary), you don’t see text garbage and you get to a Unix prompt, then your account has not been set up for PPP. Ask your provider to fix it. This is a common mistake for overworked providers.
- Reinstall MacTCP from scratch. Hey, it’s easy, but you’d be surprised how much it helps.
- Make sure your domain name server information in the MacTCP configuration dialog looks something like this (it will be different for your domain if you don’t use EarthLink):

```
earthlink.net.    198.68.160.2    (■)
.                 198.68.160.2
.                 199.2.252.10
```

- Try turning off Virtual Memory, if you’re using it, or disabling RAM Doubler, if you’re using that. One person reported success without Virtual Memory enabled.

- Ask your provider if other people are experiencing this problem as well, since it may in fact be related to an overloaded domain name server. They may be able to use a different machine for DNS services and solve the problem that way.
- Try using a different telephone cord to the modem. In one case, a slightly bad telephone cord caused some line noise that caused this error.
- Okay, here's a major realization we've made (thanks to Michael Tardiff for tremendous help in figuring this out). If nothing so far has worked, check to make absolutely sure that you are using hardware handshaking with your modem, and more importantly, make sure XON/XOFF is turned off. It seems that if XON/XOFF is turned on, MacPPP may appear to log in properly, but will fail to find the Gateway address properly.

Server-addressed accounts determine the Gateway address at connect time, which is why you cannot type it into MacTCP. Nonetheless, if, in MacPPP you don't properly use hardware handshaking, then some sort of initial negotiation that determines the Gateway address fails. When that happens, a necessary line of communication with the host is broken. The upshot of this is that MacTCP Watcher will find no Mac name for your Mac. That Mac name should be supplied by reverse name mapping, but it seems that if that initial negotiation after the login process fails, the Mac receives no Mac name from the host via the gateway. Using exactly the same settings otherwise, we showed that merely toggling a modem setting that turned hardware handshaking on and off could make the difference between a successful connection and a failed one.


- Make sure that your modem cable is indeed a hardware handshaking cable. Most high-speed modems purchased in "Macintosh kits" within the last few years include proper hardware handshaking cables, but if you bought a modem without a cable, or you bought a new cable separately from a computer store, your cable may not be a proper hardware handshaking cable. You may need to call your modem vendor to confirm this. Modern hardware handshaking cables should have the pinouts shown in Table 11.1. You can test this if you have the proper electronic testing equipment, or you can construct an electric testing device that turns on a light or makes a noise when a circuit is closed by touching leads to the proper pins (I once used an empty battery-powered squirt gun to indicate when a circuit was completed).


Table 11.1
Hardware Handshaking* Cable Pin-out

<i>Mac Function</i>	<i>RS-232 Function</i>	<i>Mac Pin</i>	<i>DB-25 Pin</i>
RxD (receive)	Receive Data	5	3
TxD (transmit)	Transmit Data	3	2

Table 11.1
continued

<i>Mac Function</i>	<i>RS-232 Function</i>	<i>Mac Pin</i>	<i>DB-25 Pin</i>
Ground	Ground	4 & 8	7
HSKi	CTS	2	5
HSKo	RTS & DTR	1	4 & 20
GPi	CD	7	8

- If you do not have a hardware handshaking cable, you can purchase them from numerous sources, but insist on a hardware handshaking cable, not just a modem cable. Jump up and down, yell, and scream, if necessary, but the two terms are not necessarily interchangeable. I've ordered high-quality, properly wired cables for about \$15 from the Celestin Company, 800/835-5514, 360/385-3767, 360/385-3586 (fax), celestin@olympus.net
- Once you determine that your modem cable is indeed a hardware handshaking cable, next ensure that hardware handshaking is turned on in MacPPP, via a pop-up menu listing choices such as XON/XOFF, CTS, RTS, and CTS/RTS. Choose CTS or CTS/RTS for hardware handshaking. Hardware handshaking is usually relatively easy to check and set.
- Unfortunately, simply turning on hardware handshaking in MacPPP may not be sufficient. It seems that some modems set XON/XOFF, or software handshaking, in their default settings. Thus, when you initialize the modem with the factory default initialization string, you actually turn software handshaking back on. This is bad. I ran into this with a Telebit QBlazer, which defaults to software handshaking with the setting S58=3. By changing that to S58=2 in the initialization string, the domain name server errors disappeared immediately. Consult the fine print in your modem's manual for the settings to ensure that hardware handshaking is enabled. Look for settings called DTE Flow Control. In Telebit modems, those settings are controlled by the S58 register, although other modems undoubtedly differ.
- Try using another modem. This worked for one user who switched from a Zoom 2400 bps modem to a QuickTel v.32bis modem. This may well have been related to the hardware handshaking issue.
- Try booting from another hard disk, if you have one.
- Try installing MacTCP, MacPPP, and the other software on another Macintosh, if you have one.
- Try rebuilding your desktop. I have no idea why this would make a difference, but it's worth a try. You can rebuild your desktop by holding down the  and Option keys during startup until the Macintosh asks you if you want to rebuild the desktop.

- Try zapping your PRAM (Parameter RAM—it stores various low-level settings). I have no idea whether this makes any difference, but anything is worth a try at this point. To zap your PRAM in System 7, hold down -Option-P-R while restarting the Mac. You will lose certain settings such as the time and the mouse speed setting, but you can reset those later.
- Try reinstalling your System. First, make sure you have a full set of System disks, since you must disable the System on the hard drive before installing, to ensure a completely clean install. Then, drag your System file to your desktop and restart with the Install disk. Install the System and then restart with your hard disk again. Throw out the System file sitting on your desktop, completely reinstall MacTCP (throwing out MacTCP DNR and MacTCP Prep), and MacPPP (throw out PPP Preferences along with the PPP extension and Config PPP). If any updates from Apple exist for the version of the System you use, reinstall them as well. Then, try again. At least one MacPPP user solved the problem this way.
- Try a different provider.
- If none of these suggestions help, I don't know what to say, except that some problems are never solved. Otherwise, we'd have world peace.

MacPPP Q & A

Although its interface can seem a bit confusing, MacPPP is actually quite easy to configure and use. But, there are still some questions.

- Q: I want to switch from InterSLIP to MacPPP. What should I watch out for?
- A: First, you need a new account. Second, I don't recommend leaving the InterSLIP extension installed at the same time as the PPP extension. Third, reinstall MacTCP from scratch before trying to connect with MacPPP. Frankly, if everything works fine with InterSLIP, I don't recommend bothering to switch.
- Q: I have an internal modem in my PowerBook, but there's no choice in the Port Name pop-up menu for Internal Modem. What should I select?
- A: Select the Modem port. Non-bus internal modems in some PowerBooks (typically the 100-series) connect to an internal modem port connector. Bus modems like the Apple Express Modem and the Global Village PowerPort/Mercury for the PowerBook 500-series and the PowerBook Duo should be set to Internal Modem.
- Q: I have an internal modem in my PowerBook 520, but I want to use an external one on the Printer-Modem port. Why doesn't that selection show up in the Port Name pop-up menu?
- A: In the PowerBook Setup control panel, there's a setting on how the modem should be treated. If it's set to Compatible, your Printer-Modem port won't appear in the Port Name menu. Set the PowerBook Setup control panel's Modem setting to Normal and all will be well.

Q: Why do I get an error message from MacPPP complaining about having insufficient memory?

A: It's been suggested that if you installed MacPPP with SAM active, and used the "allow" feature in SAM, that MacPPP and possibly MacTCP would be corrupted. The fix is to turn SAM off and reinstall both MacTCP and MacPPP from scratch.

Q: My Open button is grayed out so I can't click on it to connect.

A: The first, and most likely possibility is that you have not selected the PPP icon in the MacTCP control panel. Make sure it's selected (try selecting another icon in MacTCP then reselect PPP), and try again.

There are some known conflicts that might cause this. One possible conflict might be with an older version of the shareware SpeedyFinder, although the most recent versions solve this. Also, the elderly screensaver called Moire has been implicated in this problem as well. Finally, many Performas came with 2400 bps Global Village TelePort/Bronze modems, and if you upgrade to a faster modem, you should move the Global Village Toolbox file out of the Extensions folder and the GV TelePort Bronze file out of the Control Panels folder before trying again. And, if you have the SLIP extension from VersaTerm SLIP, InterSLIP, or MacSLIP installed, that may cause the problem as well.

Also, be very wary of any fax or remote control software that may have taken over the modem. Even if it doesn't conflict directly with MacTCP or MacPPP, the fax software may leave the modem in such a state that MacPPP is unable to access it.

Q: My Open button is still grayed out.

A: Reinstall MacTCP from scratch, making sure to delete MacTCP DNR and MacTCP Prep. This one once threw me for an hour.

Q: I still can't click on the Open button.

A: OK, one last possibility. Try reinstalling your System from scratch. Reinstalling the System seemed to help a number of users (mostly using Macintosh Performas) with this problem. First, make sure you have a full set of System disks, since you must disable the System on the hard drive before installing, to ensure a completely clean install. Then, drag your System file to your desktop, and restart with the Install disk. Install a new System, and then restart with your hard disk again. If any updates from Apple exist for the version of the System you use, reinstall them as well. Throw out the System file sitting on your desktop, then completely reinstall MacTCP (throwing out MacTCP DNR and MacTCP Prep as well) and MacPPP (throw out the PPP extension and the Config PPP control panel), and try again.

Q: When I click on the Open button, MacPPP never even dials out, but times out waiting for OK during the checking for modem phase. Do I have a bad modem init string?

A: This timeout is independent of the modem initialization string, since MacPPP isn't even getting that far. The problem is probably related to the flow control or port speed settings. Try different settings, and make sure you don't select 14,400 or 28,800 for the port speed, since some modems can't handle those as port speeds.

Also, it's possible that you have a bad modem cable, although you would probably have noticed that before.

Q: Nice try, but I still can't get MacPPP to recognize my internal Supra modem.

A: Ah, that's different. Larry Blunk mentioned a problem with the Supra and perhaps some other internal PowerBook modems. It seems that they are normally in a low-power state and can take a few seconds to warm up when MacPPP opens the serial driver and tries to dial out. Because the modem is warming up, it ignores the first initialization string that MacPPP sends, and MacPPP doesn't currently try again. The workarounds include using the terminal window or trying to script the modem initialization string in the Connect Script dialog. Larry says he's fixed this in an as-yet-unreleased version of MacPPP.

Q: Why can't I type into the terminal window?

A: In some cases, the terminal window doesn't display the characters you type (this may be related to local echo settings in the modem initialization string) until you press the Return key. Typing blind is a pain, but you shouldn't have to use the terminal window much.

Q: For some reason, I can't connect using port speeds over 19,200 bps? Why would that be?

A: Some modems don't work well at speeds over 19,200 with their default configurations. You may need to twiddle with the modem initialization string to get them to work. Don't worry about it too much though, since 19,200 bps is fast enough for a 14,400 bps modem.

Also, activating the FAXstf 3.0 LineManager option, according to one reader, prevents MacPPP from working with the SupraFAXmodem v.32bis at speeds above 19,200. In general, distrust fax software.

Q: Why can't I connect using 14,400 bps with my 14,400 bps modem or at 28,800 bps with my 28,800 bps modem?

A: You're confusing modem speed and port speed. The port speed setting in MacPPP should always be set faster than the modem speed to take advantage of modem compression capabilities. Some modems don't recognize 14,400 or 28,800 bps as valid port speeds and won't work at all.

Q: My modem will connect, and MacPPP will get to the Establishment phase, and then the Mac crashes. What could be happening? (I can use the terminal window to login fine.)

A: One reader reported a problem like this. It seems that the remote system sent a “banner page” of all sorts of text after accepting the password, but before starting PPP. MacPPP saw the banner page and got confused, because it was expecting PPP code at that point. The solution is to send several \d delay tags in the Connect Script after sending the password; that gives MacPPP some time to ignore the banner page before starting PPP.

Q: My modem will connect, and MacPPP will get to the Establishment phase, and then I get a Link Dead message. Why?

A: The most likely reason is that something in the Connect Script is wrong. You may have entered the wrong userid or password, or entered them incorrectly. It’s also not uncommon for a provider to make a mistake in setting up an account such that it’s not a PPP account, or perhaps such that it links to a different userid or password.

Also, see the previous question and answer, since sometimes the problem above doesn’t result in a crash, but merely a Link Dead message.

Finally, some providers give different userids and passwords for dialing in than they do for email, say, or different userids and passwords for shell accounts versus PPP accounts. Double check to make sure you’re using the correct ones.

Q: Sometimes when I’m trying to connect and MacPPP has to redial, I get a failure dialog with a Retry button. What should I do?

A: Click on it. I’m not positive of the circumstances in which this dialog appears, but I always just click on Retry and eventually it connects.

Q: I use a Global Village PowerPort with my PowerBook 500-series Mac, and although it works fine the first time I dial out in any session, on subsequent tries it doesn’t seem to be working. It won’t show the menu status information, nor will it make any sounds. What’s wrong?

A: This appears to be a problem exclusive to the PowerPort modems in the PowerBook 500-series. The problem lies in the Global Village software, and is actually only cosmetic, although it’s easy to get impatient and assume it’s not working. Check for a later version of the Global Village software in:

<ftp://ftp.globalvillag.com/pub/software/>

Q: MacPPP seems to redial the phone randomly on its own.

A: Click on the Hard Close button, rather than on the Soft Close button, to disconnect. The drawback to doing this is that you cannot use MacPPP’s auto connect feature after this unless you restart. Also, set a short idle timeout in Config PPP so it hangs up relatively quickly if MacPPP does dial on its own.

Make sure you don’t have any MacTCP-based applications or control panels (like Network Time) set to launch during startup. They’ll force MacPPP to dial out every time.

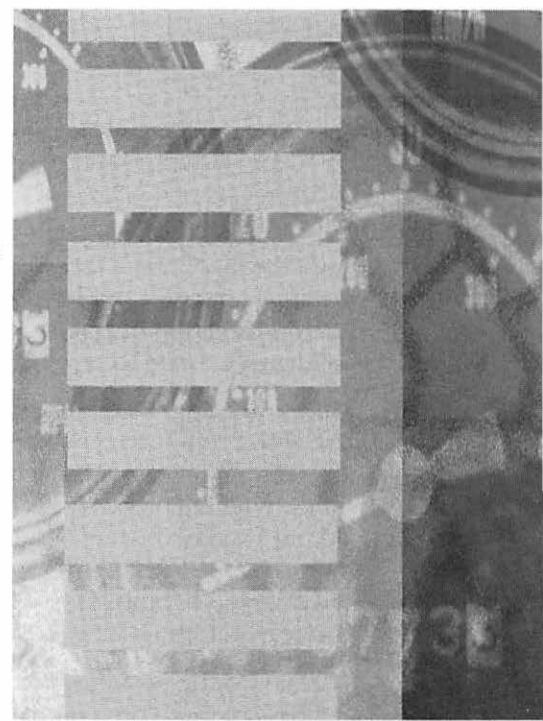
- Q: I can connect properly, but I experience very slow transfer rates in Anarchie. It seems OK at first, but gets progressively worse until the connection is basically dead.
- A: First, if you have the port speed set to 57,600 bps, try setting it down to 38,400 bps or 19,200 bps, especially if you're using a slower Macintosh with a fast modem.
- Second, I've seen this happen with MacPPP and with MacSLIP, but in both cases, I was able to solve the problem by turning off all unnecessary extensions and control panels. The hard part, then, was isolating which of them were actually causing the problem, and in some cases, more than one did. On my parents' LC II running MacSLIP, the APS PowerTools CD driver software turned out to be the culprit. On my PowerBook 100 (this was an embarrassing problem that haunted me for months), the problem was caused by PBTools 2.0, a PowerBook utility. Another user reported problems with SuperClock and the Spirit CD control panel. In general, I'd look for any control panel or extension that is in constant use, like a clock, a battery-monitoring utility, or a CD driver control panel (two others of which, the MindLink CD driver and the Apple CD driver, have been implicated in other performance problems as well).
- Q: Any idea why MacPPP crashes when I click on the Hard Close button?
- A: No, but one reader who was experiencing this tracked it down to a conflict with SAM, the anti-virus program. If you're running SAM or any other anti-virus software, try removing it first. Then, if that doesn't work, try shutting off all unnecessary extensions and control panels and see if that makes a difference. If it does, work your way back up to a full set and see which file is the culprit. I personally use and recommend only Disinfectant for anti-virus purposes.

Moving On

It's been fun, I'm sure. Once we looked at the low-level software you use to connect, MacTCP and MacPPP, we tracked down and locked up tons of problems, and I hope that any culprit that you might encounter is among them. Perhaps the most important thing to do when troubleshooting is to remain calm and proceed methodically. If you do that, you're well on your way to finding and eliminating the problem.

Actually, the real fun comes in the next chapter, where I talk about all of the programs that rely on the connection that you've established with MacTCP and MacPPP.

Chapter 12



MacTCP-based Programs

Finally! You probably never thought you would actually get to read about the MacTCP-based software that I've included on the disk for you to use. We begin with Eudora, the preeminent email program, then discuss NewsWatcher, one of the best Usenet newsreaders for the Mac. After that, we move to Anarchie, possibly the best FTP program for any computer, and MacWeb, a small yet powerful World Wide Web browser. Of course, there are many more programs than I discuss here—check out chapters 21, “Email,” through 28, “Integrated Programs,” on the version of *Internet Starter Kit for Macintosh* on the CD for more information about those other (and lesser) programs. Let's get started, but first, I want to talk about a program that a number of the others use to store their configurations, the public domain Internet Config from Peter Lewis and Quinn.

Internet Config

The Internet has a problem. There are simply too many details for anyone who doesn't spend all day using the Internet to remember. Pop quiz! What's your SMTP server? What's your NNTP server? The URL for your home page?

Sorry about that. I even promised early on that there wouldn't be any quizzes, but the point I want to make is that there are many pieces of information that not all of us necessarily remember, and there's nothing worse than trying to configure a new program and not remembering the name of your SMTP server or something equally mundane. And besides, it's a pain to type the same information into each program.

A program from Peter Lewis and Quinn "The Eskimo!" has started to solve this problem and will continue to cement the Macintosh's position as the preeminent Internet client platform. Internet Config stores all your common Internet preferences in a single place, simplifying the process of configuring MacTCP-based programs with information such as your preferred email address, FTP helper application, and helper applications. Before Internet Config, configuring all the programs with the same information was almost as bad as going to multiple doctors to have health care committed on you, given that each doctor asks for the same information on a different forms.

Internet Config provides a simple interface for setting these preferences and makes a database of those preferences available to other applications. In other words, after you enter your email address into Internet Config, both Anarchie and NewsWatcher can read it from the Internet Config database, and do not force you to enter it again and again. This capability is so useful that I've included Internet Config on the ISKM disk.

Internet Config manages the following groups of preferences:

- Personal: such as your real name and your signature
- Email: email address and other mail related details
- News: news server and related details
- File Transfer: download folder and preferred archive sites
- Other Services: default hosts for other services, like Web and Gopher
- Fonts: preferred font settings for lists, screen, and printer
- File Types: for mapping extensions to Macintosh file types
- Helpers: for mapping URLs to their helper applications

Internet programs must support Internet Config—there's no way for them to know about the preferences database otherwise. Luckily, the Internet Config development mailing list included most of the Macintosh Internet developers, and many of them have committed to supporting Internet Config in future versions of their programs. Programs that support Internet Config now include Peter Lewis's Register 1.1, NewsWatcher 2.0b27, NewsHopper 1.1, and Anarchie 1.5 (the last two of which rely entirely on Internet Config). Applications slated to support Internet Config in the future include MacWeb, InterCon's TCP/Connect II, Aladdin's StuffIt family, and NCSA Telnet.

Keep in mind that Internet Config is there to help you, not to make your life miserable. You do not have to fill in every preference immediately, or even at all.

Internet Config: Step-by-Step

Quick Reminder: Internet Config is a program that helps you configure MacTCP applications. You enter configuration information in Internet Config once, and then any MacTCP application that knows about Internet Config automatically uses the information in Internet Config. Of the programs mentioned in this chapter, Internet Config works with NewsWatcher and Anarchie.

Tasks: Configure Internet Config

1. Double-click the Internet Config icon.

Internet Config launches. If this is the first time you have launched Internet Config, it prompts you to install the Internet Config extension. Click the OK button to install it.

Next, Internet Config shows the Internet Preferences window (see figure 12.1). If for some reason you don't see the Internet Preferences window, choose Open Internet Preferences from the File menu.

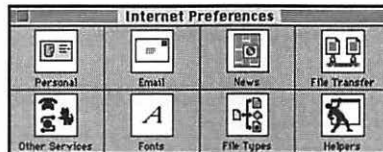


Figure 12.1 Internet Preferences window.

2. In the Internet Preferences window, click the Personal tile.

The Personal window opens (see figure 12.2).

Figure 12.2 *Setting personal preferences.*

3. In the Personal window, fill in your name and organization. Leave the > character in the Quote String field. When you finish, click the close box at the upper left of the window.

Note

For more help with any Internet Config dialog, turn on balloon help (from the Help menu at the upper right of the menu bar) and point at the item you want help with.

4. Back in the Internet Preferences dialog, click the Email tile.

The Email window opens (see figure 12.3).

Figure 12.3 *Setting email preferences.*

5. Fill in the Email Address field with the email address given to you by your provider. Be careful to use all lowercase.
6. Press Tab or click in the Mail Account field. Type your mail account, using information your provider gave you. If you have a POP account, use your POP account in the Mail Account field. Be certain to use all lowercase unless your provider explicitly gave you a mixed-case username.

7. Press Tab or click in the Mail Password field. Enter your password, being careful to enter it exactly as your provider gave it to you.
8. Press Tab or click in the SMTP Host field. Type your SMTP host, using the name your provider gave you. (If your provider gave you information about an SMTP server, note that an SMTP server is the same thing as an SMTP host.)

Note

Figure 12.3 shows my information, and yours is definitely different. Also, some providers provide different passwords for different things; be sure to put your email password in the Mail Password field.

9. When you finish filling in the first four fields in the Email window, click the close box at the upper left of the window.
10. You should be back at the Internet Preferences window. Click the News tile.

As you probably expect, the News window opens (see figure 12.4).

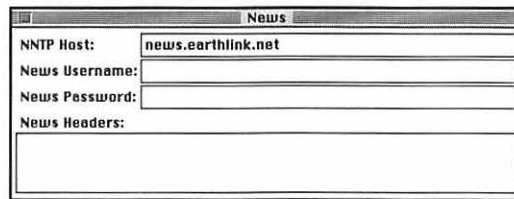


Figure 12.4 Setting News preferences.

11. In the News window, fill in the NNTP Host field using information from your provider. If your provider is EarthLink Network, enter **news.earthlink.net**, as I did in figure 12.4.

Note

If you are in Los Angeles and decide to use Leonardo Internet instead of EarthLink Network, enter **news.leonardo.net** instead.

12. Click the close box at the upper left of the window.
13. From the File menu, choose Save.

A Save dialog box appears, and it shows that if you click the Save button, you will save a file called Internet Preferences in your Preferences folder.

14. Do not change the default name or folder. Do click the Save button.
15. From the File menu, choose Quit.

You have finished the basics of setting up Internet Config. When you launched Internet Config back in step one, the program installed an Internet Config extension and an Internet Preferences file. You can also customize the settings in the remaining Internet Config tiles, but it's not necessary at this time.

Internet Config Details

Peter and Quinn have placed Internet Config and its source code in the public domain, and they encourage others to build on it to provide additional functionality. Internet Config can play a huge role in making the Mac an even better Internet platform because it can make coherent the often confusing process of configuring many different programs.

The official support address for Internet Config is `internet-config@share.com`. If you find a bug in Internet Config, forward details to that address. To discuss Internet Config in general, the `comp.sys.mac.comm` newsgroup is the best place to do so, since it allows programmers to stay in touch with the discussions without being overwhelmed with email. Once again, kudos to Peter and Quinn for a job well done. You can retrieve the latest version of Internet Config, currently at 1.1, from all the main Internet FTP sites and from the URL below.

<ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/>

Eudora

Considering that email is the ubiquitous application on the Internet, you should use the best email program available; otherwise, you will slowly (or quickly, in my case) go stark raving mad. I've looked at many email programs in my time, and although a number of them are becoming more and more impressive, none compete with Steve Dorner's Eudora. Simply put, Eudora does most everything right. Again, I don't want to imply that other programs aren't good, but none I've seen can match the features and capabilities of Eudora.

Steve first wrote Eudora while working at the University of Illinois. Because of its academic heritage, Eudora was made freely available on the Internet. Because of its clean interface and full feature set, Eudora rapidly became the Internet email application of choice. In July of 1992, Steve left the University of Illinois and went to work for a company called Qualcomm, where he continued to enhance Eudora. Because Steve and Qualcomm wanted to give something back to the educational community and taxpayers who made Eudora possible, and because free software is the best advertising for commercial software, Eudora has remained freeware. Qualcomm also has released a commercial version of Eudora that adds some nice touches and features that are essential for email users like me who get a ton of mail every day.

Note

To answer the question that almost everyone always asks, Steve named his Post Office Protocol program “Eudora,” after Eudora Welty, the author of a short story he had read, called “Why I Live at the P.O.”

The freeware version will continue to exist and will be developed in conjunction with the commercial version, but it is unlikely to receive many new features, other than those Steve deems necessary for basic email usage. For example, he added support for MIME (Multi-purpose Internet Mail Extensions), an Internet standard for transferring non-textual data via email, and support for Apple events, so that Eudora can work more closely with other programs on the Macintosh. In addition, the latest versions of both the freeware and commercial versions now work in Power Mac-native mode, significantly increasing speed.

Note

The version of Eudora 1.5.2 installed for you is not the Power Mac-native version. You can find it in either of the following:

`ftp://ftp.qualcomm.com/quest/mac/eudora/1.5/eudora152fat.hqx`

`ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/mail/`

The commercial Eudora 2.1 is extremely similar to the freeware Eudora 1.5.2. It looks about the same, and for the most part, works the same. Perhaps the most apparent additional feature in Eudora 2.1 is the filtering feature. It lets you annotate the subject of messages, change their priority, or put them in specific mailboxes based on information in the headers or the bodies of the messages. You can have as many filters as you want, and they can apply to incoming, outgoing, or selected messages. In addition, 2.1 also supports spell checking in messages via the Word Services suite of Apple events. You can even buy Eudora along with Spellswell, an Apple event-aware spelling program.

Note

Using Eudora to transfer files in email back and forth between Macs and PCs works well if your recipient either uses PC Eudora or another MIME-compatible mail program. If you're sending the files from the Mac, use AppleDouble encoding. If you're sending from PC Eudora, choose MIME before attaching the file. Both versions of Eudora automatically recognize MIME attachments and decode them automatically upon receipt.

Other useful features that exist only in Eudora 2.1 include uuencode support, automatic opening of attachments encoded in MIME, BinHex, or uuencode, multiple nickname files for organizations, support for System 7 drag and drop for attaching files to outgoing messages, stationery for frequently sent messages, menu-sharing for Frontier users, and multiple signatures. In my opinion, if you use Eudora heavily, as I do, Eudora 2.1 offers an extremely attractive set of features above and beyond the basic set in Eudora 1.5.2. For those just starting out, try Eudora 1.5.2 for a while, and if you decide you like it, consider purchasing the full Eudora 2.1 version for \$65. I can't recommend the commercial version of Eudora highly enough. And frankly, I encourage people to buy it, sending the message to Qualcomm that the community appreciates free software and is willing to support commercial versions to keep free versions available. However, for the following discussion, I'll concentrate primarily on 1.5.2 because that's the version on the CD.

Eudora: Step-by-Step

Quick Reminder: Eudora is an email client program that enables you to send and receive electronic mail.

Tasks:

1. Launch and configure Eudora
2. Compose and send an email message to President Clinton
3. Subscribe to the *TidBITS* mailing list
4. Read, reply to, and delete an email message

Launch and Configure Eudora

1. Double-click the Eudora icon.

Eudora launches.

2. From the Special menu, choose Settings.

Eudora presents you with the Settings dialog. Notice that the dialog has icons running down the left side, and clicking an icon brings up controls for that icon's options (see figure 12.5).

3. If you keep the Getting Started icon selected, you should see the settings for Getting Started, as shown in figure 12.5. In the POP Account field, enter your POP account (get this information from your provider) and be certain to enter it in exactly the same case as your provider used, usually lowercase.
4. Press Tab or click in the Real Name field. Enter your real name as you would like it to appear in your email messages.

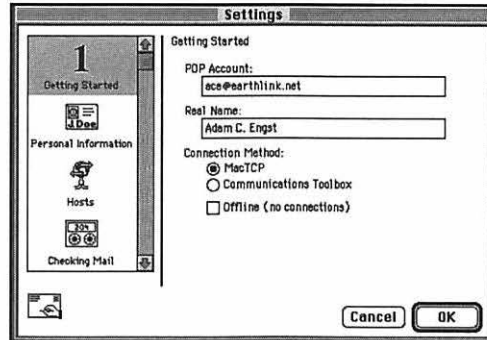


Figure 12.5 Getting started in the Eudora Settings dialog.

5. In the Connection Method area, turn on the MacTCP radio button.

You have now completed filling in the Getting Started portion of the Eudora Settings dialog. We are going to skip the Personal Information option and move on to the Hosts settings.

6. Click the Hosts icon.

The Settings dialog shows the controls for Hosts (see figure 12.6).

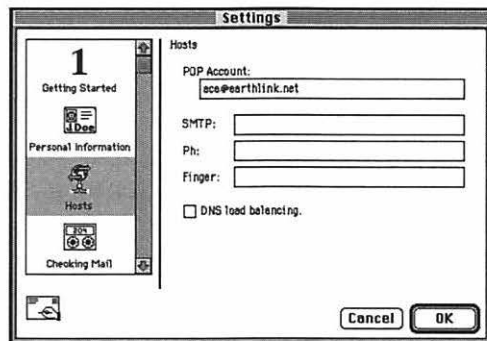


Figure 12.6 Setting Hosts options.

7. The only field you need to worry about in Hosts is the SMTP field. If your provider explicitly gave you an SMTP server (or host) that is different from the machine name in your POP account, enter its name in the SMTP field. Otherwise, leave the field blank, as I've done in figure 12.6.

That's it for the Hosts controls. Let's move on to the Sending Mail controls.

8. Click the Sending Mail icon (you may need to scroll down to see it).
The Sending Mail controls appear (see figure 12.7).

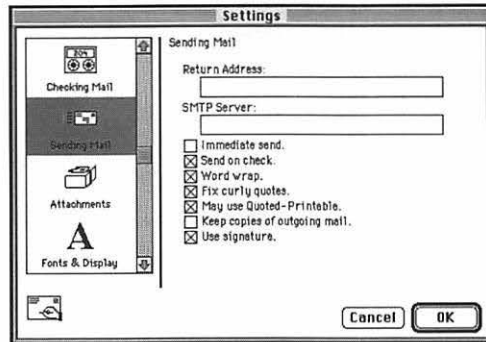


Figure 12.7 Setting Sending Mail options.

9. Turn off the Immediate send checkbox. This ensures that you can compose mail and queue it for sending without being connected to the Internet the entire time.
10. Make sure the Send on check checkbox is turned on. This ensures that Eudora sends waiting mail when it checks for new mail.
11. Click the OK button.

You have now performed the minimum configuration to use Eudora. There are many other options in the Settings dialog that you may wish to explore further. I recommend that you turn on Balloon Help from the Help menu (at the upper right of the menu bar) and point at any fields or checkboxes with which you need help.

Compose and Send an Email Message to President Clinton

1. Make sure Eudora is running. From the Message menu, choose New Message.
Eudora presents you with a new message window, with the From line filled in with your email address and name (see figure 12.8).
2. Make sure your insertion point is in the To line (it should be unless you've clicked elsewhere in the window) and type your recipient's email address. In this case, enter **president@whitehouse.gov**.

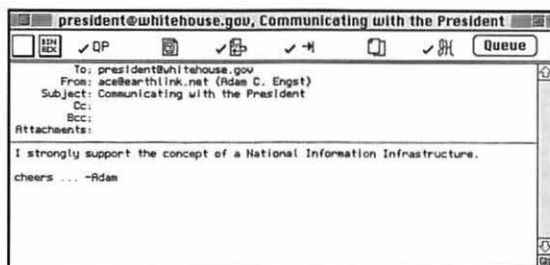


Figure 12.8 Sending email to the President.

3. Press Tab or click in the Subject line to move the insertion point to the Subject field. Enter your subject, something like *Communicating with the President*.
4. Click in the large area of the window for typing the body of your message, or press Tab three times to move the insertion point. Type your message.

Since this example sends email to an address that replies automatically, the body of the message isn't that important for the time being, although you can use this method to express your opinions to President Clinton. At minimum, type something like *I strongly support the concept of a National Information Infrastructure*. It's considered polite to sign your name at the bottom.

5. When you finish typing and signing your message, click the Queue button in the upper right corner of the window. (If that button is labeled Send, choose Settings from the Special menu, scroll down to select the Sending Mail icon, turn off the Immediate send checkbox, and click the OK button. The Send button should turn into a Queue button.)
6. Now connect to the Internet. Do not quit Eudora; simply switch out to the Finder to open Config PPP to connect.
7. Switch back to Eudora, if necessary.
8. From the File menu, choose Check Mail.

Eudora immediately presents you with a dialog asking for your password. Enter it, making sure to capitalize it as you did when you originally created it (or as it was given to you). The characters will not be displayed.

9. Click OK to enter the password you just typed.

Eudora then contacts your POP server and looks for new mail, transferring it back to your Macintosh if you have any. After retrieving new mail, Eudora contacts the SMTP server and sends the mail that you just queued for delivery. After it finishes sending, Eudora displays a dialog telling you whether or not you have new mail.

10. If you're paying for your Internet connection by the hour, or if you're paying for a long-distance call, switch to Config PPP and disconnect to save money. Otherwise, go ahead and stay connected as we work through the next few tasks.

Assuming everything was set up correctly on your Macintosh and on your host machine, you've just sent an email message via Eudora.

Subscribe to the *TidBITS* Mailing List

1. Make sure Eudora is open, and from the Message menu choose New Message.
Eudora presents you with a new message window, with the From line filled in with your email address and name and with the insertion point in the To line.
2. In the To line, type **listserv@ricevm1.rice.edu**.
3. Press Tab four times or click in the message section of the window. Type **SUBSCRIBE TIDBITS your full name** (replace **your full name** with your real name, not your email address) and nothing else (see figure 12.9).

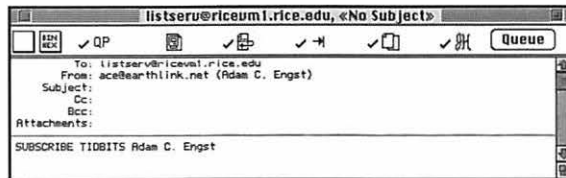


Figure 12.9 *Subscribing to TidBITS.*

4. Click the Queue button in the upper right of the window to queue the message to be sent.
5. Make sure you are connected to the Internet.
6. From the File menu, choose Check Mail.
7. As before, Eudora first connects to your POP server and checks for new mail. Depending on how long it has taken you to create this message, you may have received mail back from the White House (if you sent a message in the previous set of steps). Either way, after checking for new mail, Eudora contacts your SMTP server and sends your subscription message to the LISTSERV program.
8. If you are paying for your connection, feel free close the connection now to save money.

You've just subscribed to a mailing list! Although other mailing lists may be slightly different, mostly in terms of the mailing list manager's address and the list name, the basics are the same.

Read, Reply to, and Delete an Email Message

1. Make sure Eudora is running.

If you received a reply from the White House when you sent the subscription message to the *TidBITS* list, Eudora automatically opened your In box for you.

If you have not yet received the reply from the White House or the confirmation of your subscription to the *TidBITS* list, wait for a while (there's no way to know how long it could take, although when I wrote these instructions, the responses came back within minutes).

2. Make sure you're connected to the Internet, and from the File menu choose Check Mail.
3. Eudora opens your In box after receiving new mail; if you have closed it while waiting, go to the Mailbox menu and choose In. Eudora then displays the In box and marks unread messages with a bullet (•) character (see figure 12.10).

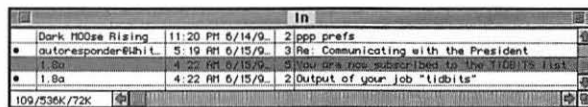


Figure 12.10 Eudora In box.

4. Double-click the reply from the White House, which probably looks as though it came from **autoresponder@WhiteHouse.Gov**, which is the program that automatically replies to email sent to President Clinton.

Eudora opens the message and displays it, along with the first four lines of the header (see figure 12.11).

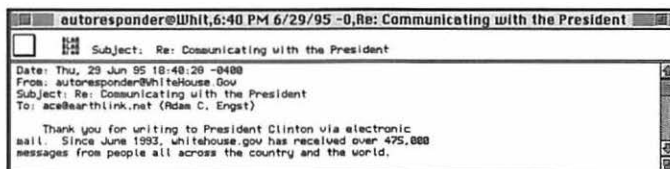


Figure 12.11 Email from the White House.

5. Read the message, scrolling with the scroll bar or the Page Up and Page Down keys.
6. After reading the message, go to the Message menu and choose Reply.

Eudora creates a new message window, entering the original sender's address in the To line and the subject of your original message, prefixed with Re:, in the Subject line. The entire body of the original message is quoted in the body of the message, and Eudora automatically selects the quoted text (see figure 12.12). You can edit this text or delete it entirely by pressing the Delete key.

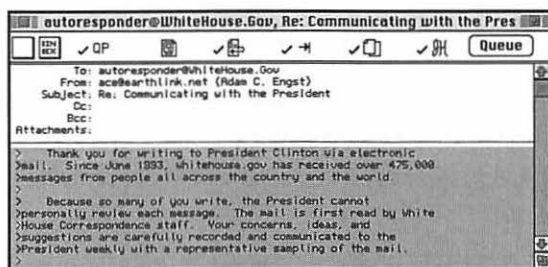


Figure 12.12 Replying to a message.

7. When your reply is ready, you could click the Queue button to queue it for delivery again, but please don't, unless you really want to send mail to the President again.

When replying to personal email, you would queue the message, and then—when you wanted to send the message, perhaps along with other queued messages—you would choose either Check Mail or Send Queued Messages from the File menu.

8. Once you finish sending any messages, disconnect from the Internet, especially if you're being charged.
9. To delete the message from the White House, make sure its window is open or make sure it is selected in the In box; then, from the Message menu choose Delete.

Eudora moves the deleted message to the Trash mailbox.

That's about all you need to know to get started reading and writing email with Eudora. As you explore the program (or read about Eudora in chapter 21, "Email," of the CD version of *Internet Starter Kit for Macintosh*) you will find many other options and shortcuts to make using Eudora even easier.

Eudora Details

Eudora 1.5.2 is free, and comes courtesy of the University of Illinois and Qualcomm. You can retrieve the latest versions from:

```
ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/mail/
```

```
ftp://ftp.qualcomm.com/quest/mac/eudora/1.5/
```

Eudora 2.1 costs \$65 for an individual copy, or \$99 for Eudora 2.1 and Spellswell. Prices drop quickly from there, depending on how many copies you want to buy, so if you're outfitting a couple of people in an office, check with Qualcomm for the exact discounts. You can get more information from Qualcomm via email at **eudora-sales@qualcomm.com**, or by phone at 800-2-EUDORA.

Although official technical support only comes with Eudora 2.1, expert Eudora users provide extremely good support online in the newsgroup **comp.sys.mac.comm**. Also, Qualcomm runs a mailing list specifically for Eudora users—to subscribe, send email to **majordomo@qualcomm.com** with **subscribe mac-eudora-forum** in the body of the message. Make sure that your return address is correct—it is the address that will be added to the mailing list. If you have a simple question, ask there before anywhere else (but use the balloon help and the manual before that). Eudora's manual is also excellent. Look for it in:

```
ftp://ftp.qualcomm.com/quest/mac/eudora/documentation/
```

On the
CD

For more information about other email programs and utilities, check out chapter 21, "Email," in the CD version of Internet Starter Kit for Macintosh.

NewsWatcher

In the realm of newsreaders for Usenet news, I currently prefer NewsWatcher, developed by John Norstad of Disinfectant fame. Unfortunately NewsWatcher is not on the CD, but you can easily download it using the Get NewsWatcher bookmarks in your Get New Internet Programs folder (the ISKM Installer installs this folder for you in the ISKM3 Folder on your hard disk). Alternately, just read on for the step-by-step instructions for Anarchie, which walk you through getting NewsWatcher.

There are several other excellent Usenet newsreaders that you might want to try as well as NewsWatcher. They are Nuntius, freeware from Peter Speck, NewsHopper, a commercial program from SW15 Software, and InterNews, which is shareware from Dartmouth College. Check out chapter 22, "Usenet News," on the CD version of Internet Starter Kit for Macintosh for more information about these other programs.

The major challenge that the newsreaders face is presenting a clean and quick method of navigating through gobs of information. Interface is all-important (and a purely personal choice, of course), but raw speed doesn't hurt either, and NewsWatcher feels fast. Steve Falkenburg of Apple first created NewsWatcher, and John Norstad later picked it up to continue development. John has made the current version, 2.0, into one of the best Internet programs available today.

NewsWatcher: Step-by-Step

Quick Reminder: NewsWatcher is a newsreader, a client program for Usenet news. NewsWatcher requires access to an NNTP news server on an Internet host (luckily, almost all Internet access providers offer NNTP access to their customers). You must download NewsWatcher from the Internet—check your bookmarks in Anarchie and the Get New Internet Programs folder.

Tasks:

1. Launch and configure NewsWatcher
2. Create a personalized subscription list and subscribe to several newsgroups
3. Read articles in a subscribed newsgroup
4. Post an article

Launch and Configure NewsWatcher

1. Make sure you are connected to the Internet—connect with MacPPP if necessary.
2. Double-click the NewsWatcher icon.

If you haven't previously configured NewsWatcher, the program launches and presents a Welcome to NewsWatcher dialog (see figure 12.13).

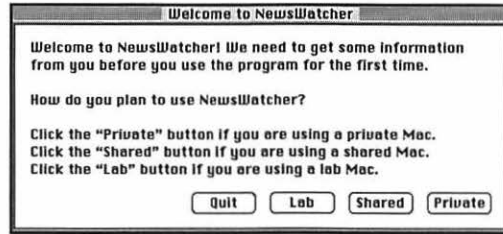


Figure 12.13 Choosing how you use NewsWatcher.

3. Click the Private button.

The Server Addresses dialog appears. If you configured Internet Config, the fields will already be filled in (see figure 12.14).

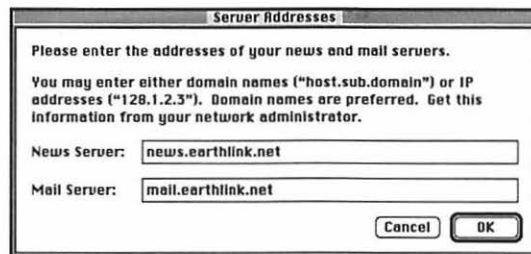


Figure 12.14 Configuring NewsWatcher.

4. If they are not already filled in, fill in the News Server and Mail Server fields using information you have obtained from your provider.
5. Click the OK button.

The Personal Information dialog appears.

6. Enter correct information in the Full name, Organization, and Email address fields. Be certain to type your email address with all lowercase letters.
7. Click the OK button.

NewsWatcher connects to your news server and retrieves the full group list from the server. This can take some time, especially over a slow modem. Once NewsWatcher sorts the list, it displays it in a scrollable window, along with a smaller window labeled untitled.

Do not quit NewsWatcher now, but go on to the next task in which you learn how to create a personalized subscription list and subscribe to newsgroups that might interest you.

You've now successfully completed the minimum steps necessary to configure and use NewsWatcher. I strongly recommend that you read my discussion of NewsWatcher in chapter 22, "Usenet News," on the CD version of *Internet Starter Kit for Macintosh*; and read the user documentation that comes with NewsWatcher. Most of the additional configuration options live in the Preferences dialog, which is available from the File menu (see figure 12.15).

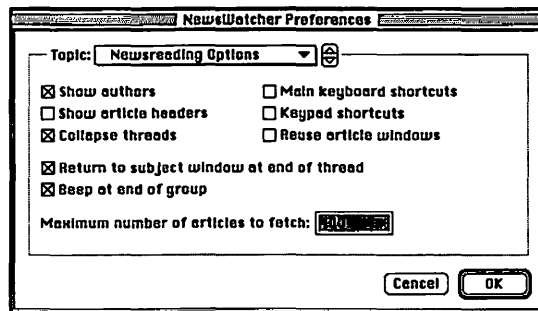


Figure 12.15 *NewsWatcher Preferences dialog.*

Create a Personalized Subscription List and Subscribe to Several Newsgroups

1. Arrange the untitled and Full Group List windows so they don't overlap.
2. Making sure that untitled is the frontmost window, go to the File menu and choose Save. Give the file an appropriate name, such as **My Newsgroups**, and save it in a location where you will be able to easily find it later.

The window, also known as the *subscription window*, takes on the name that you gave it.

You may wish to move the file later, perhaps to your Apple Menu Items folder so that it shows up in your Apple menu.

3. Scroll down in the Full Group List window until you find the newsgroup called **news.announce.newusers**. Groups are sorted alphabetically, so it should be about halfway down.
4. Drag the **news.announce.newusers** item over to the subscription window and drop it in the window.

NewsWatcher may show the spinning beach ball cursor briefly, and **news.announce.newusers** should appear in your subscription window. The number next to its name indicates the number of unread articles in that group.

- Repeat the process with **comp.sys.mac.announce**, **comp.sys.mac.comm**, and **misc.test**. As you drag each name over the subscription window, you should see a dark black line appear in the window, indicating where the newsgroup will appear once you drop it. For some fun, you might also drag over **rec.humor.funny**. You can also add any other groups that you think might be interesting. When you finish, your subscription window should resemble the one in figure 12.16.

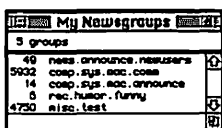


Figure 12.16 NewsWatcher subscription window.

Announcements important to the entire Mac community appear in **comp.sys.mac.announce**. Discussions about Macintosh communications software appear in **comp.sys.mac.comm**, which is also a good place to ask about things that you cannot otherwise figure out. I use **misc.test** later, when providing instructions on posting.

- Make sure your subscription window is frontmost, and choose Save from the File menu to save your subscription list.
- Close the Full Group List window by clicking the close box at its upper left. (You can open it again by choosing Show Full Group List from the Windows menu.)
- Quit NewsWatcher by choosing Quit from the File menu.

The previous isn't absolutely necessary, but bear with me. I want you to start the next task, reading articles, as you would normally do it, and that includes launching NewsWatcher.

You've now successfully created a personalized subscription list and saved it for future use. You can add newsgroups to this list at any time and remove newsgroups that no longer interest you (to remove a newsgroup, select it in the subscription window and then choose Unsubscribe from the Special menu). NewsWatcher starts up slightly faster with a small subscription list, so it works best to only subscribe to newsgroups that you read.

Read Articles in a Subscribed Newsgroup

1. If you are not connected to the Internet, establish a connection.
2. Launch NewsWatcher, not by double-clicking the NewsWatcher icon, but by double-clicking the subscription list icon that you created in the previous task (for example, I would have double-clicked on “My Newsgroups”).

NewsWatcher launches, connects to the news server, checks for new groups and new articles, and then displays your subscription list window.

3. Double-click **news.announce.newusers**.

NewsWatcher retrieves the subjects and authors of the articles contained in that newsgroup and presents you with a window displaying a list of those articles (see figure 12.17).

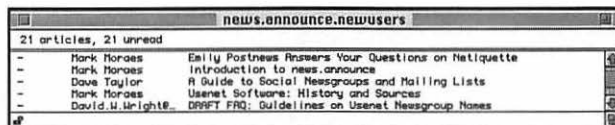


Figure 12.17 Articles in **news.announce.newusers**.

4. Double-click the first article in the newsgroup.

Since articles in **news.announce.newusers** are often relatively large, it may take a minute or so to download if you're using a modem. NewsWatcher displays the article once it is downloaded.

5. Read the article, if you wish. I recommend that you browse the articles in this group early on, since they're designed to answer many questions that new users have.
6. To read the next article, you can close the article window and double-click the next article in the newsgroup. An easier method is to go to the News menu and choose Next Article or to press ⌘-I.

You've now successfully opened a newsgroup and read several articles. You can close the window listing articles in **news.announce.newusers**, and double-click **comp.sys.mac.announce** to see the list of articles in that group and read them if you wish.

Post an Article

1. In your subscription window, double-click the **misc.test** newsgroup. You could double-click any newsgroup that you wanted to post to, but since this is your first attempt at posting from NewsWatcher, we'll try a newsgroup specially for tests. You will receive a number of email messages from various sites, letting you know that your test posting made it there.

The **misc.test** window opens.

2. From the News menu, choose New Message.

NewsWatcher brings up the New Message window with the Newsgroups line already filled in and the insertion point in the Subject line (see figure 12.18).

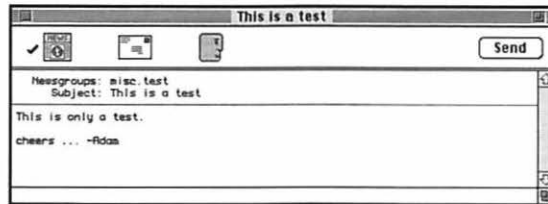


Figure 12.18 NewsWatcher new message window.

3. Type a subject in the Subject line.
4. Press Tab or click in the message area. Then, type your article.
5. Click the Send button.

NewsWatcher posts your article to **misc.test**. Now that you've posted your article, feel free to read other articles, post more articles (but remember, it's a good idea to lurk for a while before becoming a prolific poster), and subscribe to additional newsgroups. You won't see your post in **misc.test** unless you quit NewsWatcher and relaunch it.

6. When you're finished, choose Quit from the File menu and save your subscription list when NewsWatcher prompts you to do so.

Congratulations! You've successfully performed all the basic tasks you do with NewsWatcher. I won't pretend that there aren't many more subtleties in using this program, but you can learn about those from reading chapter 22, "Usenet News," in the CD version of *Internet Starter Kit for Macintosh*, and NewsWatcher's documentation files.

NewsWatcher Details

NewsWatcher 2.0 is free, and you can retrieve the latest version using either the Get NewsWatcher bookmark on your hard disk or from either of the following:

`ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/`

`ftp://ftp.acns.nwu.edu/pub/newswatcher/`

On the
CD

For more information about other Usenet newsreaders and utilities, scan to the end of chapter 22, "Usenet News," in the CD version of Internet Starter Kit for Macintosh.

Anarchie

Although more *people* use email than any other Internet application, more *data* is transferred via FTP than via any other Internet service (except maybe the Web, by the time you read this). FTP is one of the base services that tie the Internet together. Despite its relative simplicity from a command line, FTP works far better when you can use a graphical application to navigate through remote directories and files.

For most people, Peter Lewis's Anarchie is quite simply the best FTP client available for any platform. Not only is it fast, easy to use, and cleanly implemented, but it also does something that I've wanted for quite some time. Anarchie is a full-fledged FTP client, but it can also *search* Archie servers for files stored on anonymous FTP sites, and once it has found those files, it can retrieve them via FTP with merely a double-click.

Note

Anarchie is pronounced like the word "anarchy," not like the phrase, "an archie." That's the word direct from Peter.

Anarchie: Step-by-Step

Quick Reminder: Anarchie is a shareware FTP and Archie client program that you use to search for and retrieve files available via anonymous FTP. Currently, the FTP function is substantially more useful (lately, using Archie to search for files has not worked well, though this is no fault of Anarchie's), so these steps focus on using Anarchie for FTP.

Tasks:

1. Launch and configure Anarchie
2. Connect to a site and retrieve a file
3. Use a bookmark to retrieve a file

Launch and Configure Anarchie

1. Anarchie uses Internet Config for its basic configuration, so if you haven't already gone through the steps earlier in this chapter for setting up Internet Config, go back and follow them. If for some reason you don't know all the information required by those steps, the minimum that you must do in Internet Config in order to use Anarchie is use the Email tile to open the Email dialog, where you must fill in the Email Address field.
2. Anarchie's job, of course, is to download files, but because Anarchie also tries to be a fairly helpful application, it passes downloaded files off to other programs, most notably StuffIt Expander, for post-processing—making the files proper, uncompressed Macintosh files. As a result, if you don't have a copy of StuffIt Expander on your hard disk, install StuffIt Expander using the ISKM Installer before you continue.
3. Double-click the Anarchie icon.
Anarchie launches.

That's it. You're done configuring and launching Anarchie and ready to use Anarchie to connect to a site and retrieve a file.

Connect to a Site and Retrieve a File

1. Make sure that you are connected to the Internet and that Anarchie is running.
2. From the FTP menu, choose Get.
Anarchie brings up the Get via FTP window (see figure 12.19).
3. In the Machine field, enter **ftp.tidbits.com**.
4. In the Path field, enter **/pub/tidbits/select/newswatcher.hqx**.
5. Select the Get File radio button.

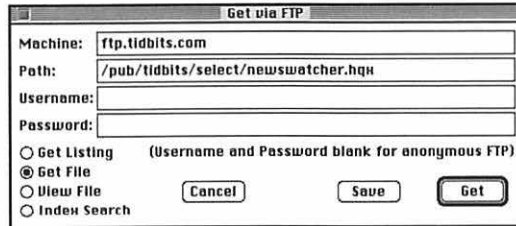


Figure 12.19 Anarchie Get via FTP window.

6. Click the Get button to retrieve the file.

Anarchie displays a progress window as it downloads the file (see figure 12.20). When the download is complete, Anarchie passes the file to StuffIt Expander for post-processing. StuffIt Expander debinhexes and expands the file, thus turning it into a NewsWatcher folder, which will appear on your desktop unless you've used Internet Config to change the location of your download folder. (You change the download folder using the File Transfer tile.)

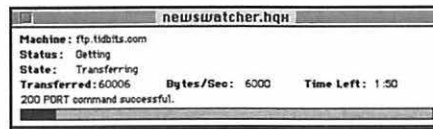


Figure 12.20 Anarchie progress window.

Use a Bookmark to Retrieve a File

1. Make sure Anarchie is running and that you are connected to the Internet.
2. If the Bookmarks window is not showing, choose List Bookmarks from the File menu (see figure 12.21).

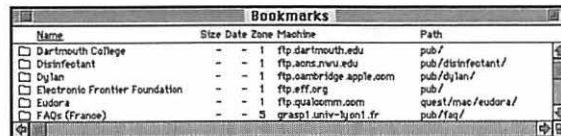


Figure 12.21 Anarchie Bookmarks window.

3. Double-click the Disinfectant item in the Name column (you may have to scroll down to see it).

Anarchie connects to **ftp.acns.nwu.edu** and puts you in the **/pub/disinfectant** directory (see figure 12.22).

Name	Size	Date	Zone	Machine	Path
announcement-36.txt	2k	4/14/95	1	ftp.acns.nwu.edu	/pub/disinfectant/announcement-36.txt
disinfectant36.sea.hqx	233k	4/7/95	1	ftp.acns.nwu.edu	/pub/disinfectant/disinfectant36.sea.hqx
README	1k	4/14/95	1	ftp.acns.nwu.edu	/pub/disinfectant/README
sample24.sea.hqx	386k	9/27/94	1	ftp.acns.nwu.edu	/pub/disinfectant/sample24.sea.hqx

Figure 12.22 Anarchie at **ftp.acns.nwu.edu**.

4. Double-click the **disinfectant36.sea.hqx** item. (The name may not exactly match, since Disinfectant may be at a later version when you try these steps.)

Anarchie downloads Disinfectant. When the download is complete, Anarchie passes the file to StuffIt Expander. StuffIt Expander debinhexes and expands the file, thus turning it into a Disinfectant icon, which will appear on your desktop unless you've used Internet Config to change the location of your download folder.

Congratulations! You've just performed all the basic tasks in Anarchie you're likely to do in real life. Extrapolate from these instructions to retrieve other files using Anarchie's Bookmarks and Get via FTP dialog.

Anarchie Details

Anarchie costs \$10. I strongly encourage you to pay for Anarchie if you find yourself using it. It's an essential Internet tool, and we need to keep Peter happy so that he keeps writing great programs and releasing them as freeware or shareware. Peter now works with Kagi Shareware, a company which accepts shareware payments in a multitude of forms for shareware authors and then pays the authors all but a small handling fee. You can use the Register application, which Peter also wrote and which comes with Anarchie, to easily pay for Anarchie through Kagi.

Anarchie is included on the disk that comes with this book, and you can retrieve the latest version of Anarchie in:

<ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/>

For more information about other FTP programs, scan to the end of chapter 23, "FTP," in the CD version of *Internet Starter Kit for Macintosh*.

MacWeb

When I wrote the first edition of *Internet Starter Kit for Macintosh*, the World Wide Web was just starting to explode and the Macintosh was being left out. Then, along came NCSA Mosaic for the Macintosh and all was well in the world. Since that time (fall of 1993), the World Wide Web has propelled the Internet into the eyes of the public. "Check out my home page at..." has become a common phrase, and the amount of information that appears every day on the Web is staggering.

Despite the fact that the Web was developed at CERN, in Switzerland, NCSA Mosaic deserves a good deal of credit for popularizing the Web. NCSA had the resources to create Mosaic and to give it away, and although it wasn't a particularly good program at that point, it was enough to entice people to use and publish information on the Web.

Of course, any success demands to be copied or even exceeded, and that's where the other major Web browsers came from. First was MacWeb, from the EINet group of MCC, and then came Netscape, developed by many of the same programmers who had created Mosaic while at NCSA. Then, NCSA, via a company called Spyglass, licensed the Mosaic code to a variety of companies that have produced versions of Enhanced Mosaic that differ little from the first versions of NCSA Mosaic. Most recently (and discussed in chapter 28, "Integrated Programs," on the CD version of *Internet Starter Kit for Macintosh*) InterCon has added a Web module to its TCP/Connect II integrated program.

For the most part, the Web browsers are extremely similar, so I'm going to start with MacWeb because it sorts first in an alphabetical listing of the browsers' names and because we've licensed it for inclusion on the ISKM disk.

The beauty of writing about Web browsers is that there are almost no instructions to give. The basic idea is that you connect to the Internet, run the Web browser, and then click on the underlined words (they can also be in a color, but that doesn't show up well in a black and white book) to traverse the links between Web pages.

That's about all there is to using the Web. We're not talking difficult here.

MacWeb has always differentiated itself by being small and quick. New features, such as a pop-up menu that appears if you click and hold on a link, often appear in MacWeb first and are then copied by the other Web browsers. Because of its small size and low memory requirements, MacWeb is the browser of choice for people with older Macs or not much memory. Just for comparison, the latest versions of MacWeb, Netscape, and NCSA

Mosaic weigh in at 470K, 1,370K, and 2,200K on disk, respectively, and MacWeb can get by in as little as 750K of RAM, compared to 1,784K for Netscape (which prefers more) and 2,700K for Mosaic. Because of these reasonable requirements and because it's a good program, we've included MacWeb on the CD.

MacWeb: Step-by-Step

Quick Reminder: MacWeb is a client application for the World Wide Web, the most graphical and flexible of the Internet services.

Tasks:

1. Launch and configure MacWeb
2. Browse the Web
3. Visit a specific web site
4. Use the hotlist

Launch and Configure MacWeb

1. Make sure you are connected to the Internet—connect with MacPPP, if necessary.
2. Double-click the MacWeb icon.

MacWeb launches and loads its default home page. Figure 12.23 shows what the MacWeb default home page will look like if you use installed MacWeb using the ISKM Installer. Don't worry if yours looks different, though you should see the same buttons across the top.



Figure 12.23 MacWeb welcome page.

3. From the File menu, choose Preferences.

MacWeb brings up the Preferences dialog and displays its General options.

4. Type your email address in the Email Address field, making certain to use only lowercase letters.
5. From the pop-up menu at the top center of the dialog, choose Format.

MacWeb displays its Format options (see figure 12.24).

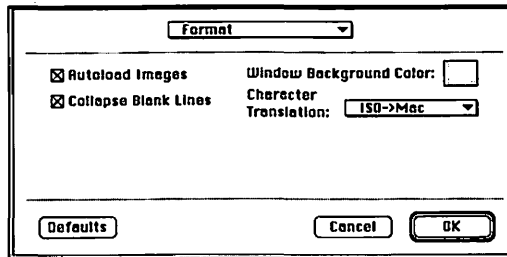


Figure 12.24 *MacWeb preferences dialog.*

6. Turn on the Autoload Images checkbox.
7. Click the OK button to save your changes.

That's all there is to it. You now have MacWeb configured and ready to go. If you get tired of automatically loading images on the Web pages you view, you can go back to the preferences dialog and turn Autoload Images back off.

Browse the Web

1. Make sure MacWeb is launched and that you are connected to the Internet.
2. From MacWeb's Navigate menu, choose EINet Galaxy.

MacWeb displays the contents of the EINet Galaxy page (see figure 12.25).

3. Scroll down until you get to the Arts and Humanities topic. Click the underlined words Visual Arts.

MacWeb takes you to the Visual Arts page on the EINet Galaxy (see figure 12.26).

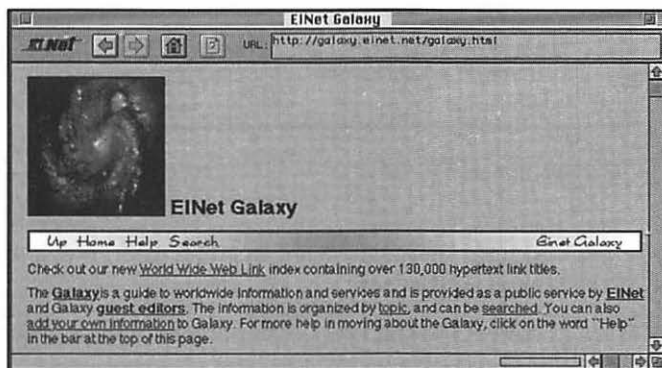


Figure 12.25 MacWeb at EInet Galaxy.

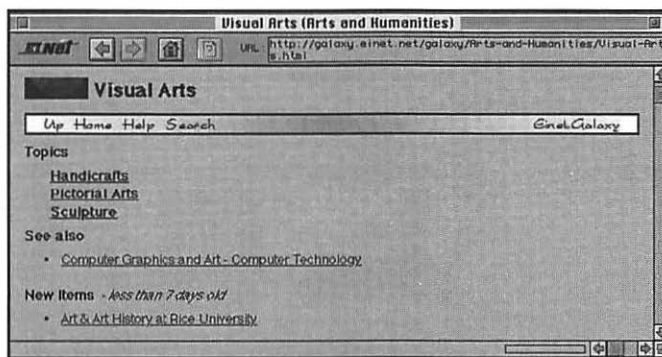


Figure 12.26 MacWeb at the Visual Arts page.

I hope you're feeling comfortable with clicking on underlined text to move around in the Web, because if you are, you know the basics of browsing the Web. Feel free to continue clicking on underlined words (which are called *links*) to move to other parts of the Web—it's too large and fast-moving for me to give you any further explicit browsing directions. If you are wondering what to do if you know a URL for a Web site that you want to visit, keep reading to find out how.

Visit a Specific Web Site

1. Make sure you are connected to the Internet and that you have MacWeb launched.
2. From MacWeb's File menu, choose Open URL. MacWeb displays a dialog into which you can type a URL (*Uniform Resource Locator*) (see figure 12.27).

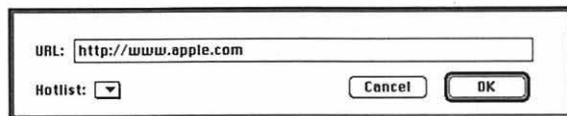


Figure 12.27 Type a specific URL.

3. In the field, type the URL for where you want to go to. To follow along with this example, type **http://www.apple.com**
4. Click the OK button to connect to the Web server at **www.apple.com** (see figure 12.28).



Figure 12.28 MacWeb at the Apple Computer Web server.

That's all there is to going to a particular URL. If you see URLs in publications, on business cards, or wherever, you can type them into the Open URL dialog or you can also paste them from other programs. You can also type or paste URLs directly into the URL box at the top right of the MacWeb window, though this takes some hand-eye coordination.

If you are thinking about taking a break, please don't. I'm hoping you'll continue to the next set of steps so that you can see how to add an entry to your MacWeb hotlist.

Use the Hotlist

1. With the window to the Apple Computer Web server still open, choose Add This Document from the Hotlist menu.

MacWeb adds the Apple Computer Web server to the bottom of the Hotlist menu.

2. To see how the hotlist works, click the Home button (the one with a little house on it) at the top of the MacWeb window.

MacWeb loads your default home page.

3. From the Hotlist menu, choose Apple Computer.

MacWeb takes you directly to the Apple Computer Web server.

4. To save your hotlist, go to the Hotlist menu and from the hierarchical Hotlist Operations menu, choose Save.

MacWeb brings up a Standard File dialog, and you can save the hotlist anywhere you wish. Remember where you save it, though, because in your next MacWeb session you may find yourself trying to open your hotlist by dropping down the Hotlist menu, choosing the Hotlist Operations hierarchical menu, and then using the Open command to open the hotlist. (Of course, you can avoid these steps by double-clicking the hotlist icon in order to launch MacWeb.)

5. In the Save dialog, name your hotlist and save it in a location where you'll be able to find it again.

That's about all there is to using MacWeb, though exploring even a portion of the Web could take a lifetime. When you finish looking around for now, quit MacWeb by choosing Quit from the File menu. You may wish to disconnect from the Internet after you quit.

MacWeb Details

MacWeb was written by John Hardin of EINet and is freeware for academic, research, or personal use. To report problems with or make suggestions about MacWeb, send email to **macweb@einet.net**. You can retrieve the current version of MacWeb on the Internet at either of the following:

`ftp://ftp.einet.net/einet/mac/macweb/`

`ftp://ftp.tidbits.com/pub/tidbits/tisk/tcp/`

Netscape Navigator

The greatest concentration of ex-NCSA Mosaic developers can be found at the Mountain View offices of startup Netscape Communications. Founded by Jim Clark, previously head of Silicon Graphics, and Marc Andreessen, who created the first NCSA Mosaic, Netscape has gone from complete obscurity to being one of the heavy hitters in the world of the Web. The reason? Netscape Navigator, which is what happens when you take talented developers and ask them to write a program they've done once all over again from scratch, avoiding the mistakes they made the first time and rethinking the parts that didn't work well. Netscape Navigator basically owns the Web browser market, if you can call it that since most Web browsers are essentially free, with some estimates giving it as much as 75 percent market share.

Netscape Navigator (generally just called Netscape, thanks to some weaseling around with the name early on, when the company was called Mosaic Communications and the program was called Mosaic Netscape) shines in two specific areas. It's fast, thanks to an innovative way of establishing multiple connections to the server when you retrieve a Web page, and it has, by far, the best hotlist feature (called bookmarks in Netscape's parlance).

Note

HTML, or HyperText Markup Language, is the language in which documents are written for display on the Web.

Actually, there's a third major reason Netscape took the Web by storm. Netscape Communications "extended" the standard HTML 2.0 tags in advance of the forthcoming HTML 3.0 specification, and supported those extensions in Netscape. The result is that people writing in HTML can do things graphically, such as wrap text next to a graphic, that were previously impossible (and still are in other Web browsers). Suddenly, if you wanted to see a page in all its glory, you had to use Netscape. Many people felt that Netscape's jumping of the gun wasn't particularly fair play, and many Web page developers refuse to use Netscape-specific HTML codes until the HTML 3.0 specification is complete (which should be sometime in 1995). At that point MacWeb and Mosaic will almost certainly add support for all the additional HTML codes that enable Web page developers to create tables, wrap text around graphics, place graphics in specific spots on the page, and so on.

Netscape Step-by-Step

Quick Reminder: Netscape is a client application for the World Wide Web, the most graphical and flexible of the Internet services. You must download Netscape from the Internet—check your bookmarks in Anarchie or go to **ftp.netscape.com**.

Tasks:

1. Launch and configure Netscape
2. Browse the Web
3. Visit a specific Web site
4. Use a bookmark

Launch and Configure Netscape

1. Make sure you are connected to the Internet.
2. Double-click the Netscape icon.

Netscape launches, and displays a dialog reminding you of its licensing agreement. (Don't worry if this dialog doesn't appear, just skip the next step if you don't.)

3. Click the Accept button.

Netscape finishes launching and connects to its default home page, which—in most cases—looks like the one in figure 12.29.



Figure 12.29 Netscape home page

4. Go to the Options menu and choose Preferences.

The Preferences dialog opens.

5. In the pop-up menu at the top center, make sure that Mail and News is chosen (see figure 12.30).

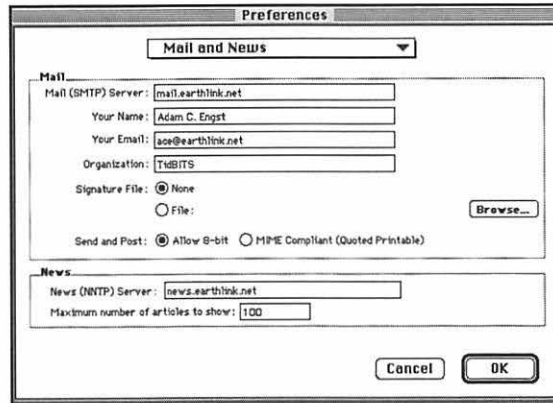


Figure 12.30 Netscape Mail and News preferences

6. Enter your Mail Server in the Mail (SMTP) Server field. Use the name that your provider gave you. (Note that a mail host is the same thing as a mail server.)
7. Press Tab or click in the Your Name field. Type your name.
8. Press Tab or click in the Your Email field. Enter your email address, as it was given to you from your provider. Make certain to type it in all lowercase.
9. Press Tab or click in the Organization field. Type your organization.
10. Click the OK button to close the Preferences dialog and save your changes.

You just finished a fairly minimal Netscape configuration. As I'm sure you noticed, there are many options in the Preferences dialog box that you didn't customize. If you want to learn more about the them, choose Handbook from the Help menu. Fortunately, you need not worry about configuring more of Netscape now, or possibly ever.

Browse the Web

1. Make sure you are connected to the Internet and have Netscape launched.
2. In Netscape, click the Net Directory button.

Netscape loads a page called Internet Directory (see figure 12.31).

Note

The Netscape Web site changes every now and again, so if you see a different page after clicking the Net Directory button, don't worry, though these directions may not work out if you can't find a Yahoo link on the page that does load.

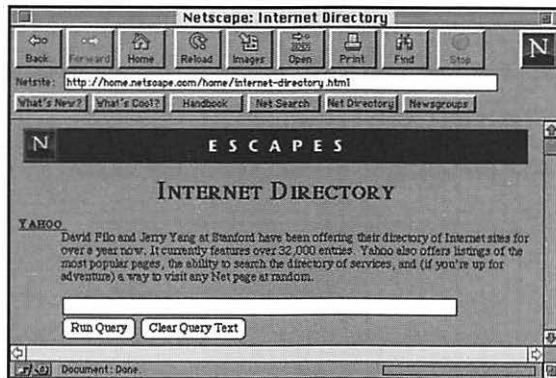


Figure 12.31 Netscape Internet Directory.

3. Click the underlined word Yahoo.

Netscape loads the home page for Yahoo, a page with lots of different links (see 12.32).

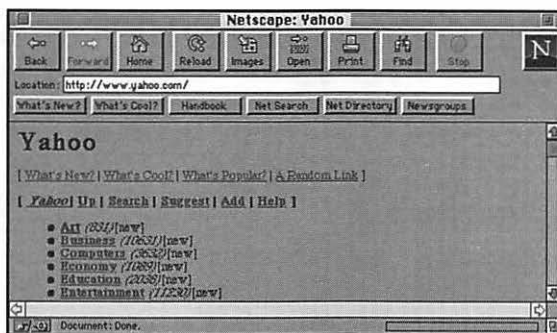


Figure 12.32 Yahoo's list of Web links.

4. Click the underlined word Art.

Netscape loads the Art page.

Please feel free to continue clicking underlined text. In just another click or two, you should get away from the lists of links and into real information. It would take years and years to explore the entire Web, but for now I hope that you have the basic idea of clicking on underlined text to move from topic to topic.

Visit a Specific Web Site

1. Make certain you are connected to the Internet with Netscape running.
2. From the File menu, choose Open Location.

Netscape opens the Open Location dialog (see figure 12.33).

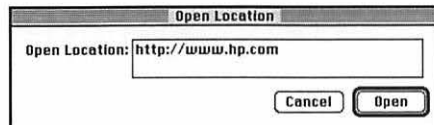


Figure 12.33 Netscape Open Location dialog.

3. Type **http://www.hp.com**
4. Click the Open button

Netscape connects to Hewlett Packard's home page (see figure 12.34).

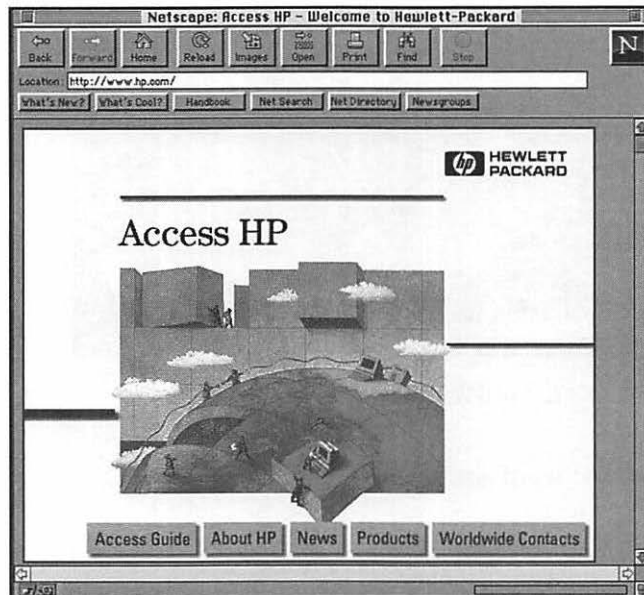


Figure 12.34 Netscape at the HP Server.

Congratulations! You now know the steps for typing in a URL to a specific Web site. If you are comfortable with the small Location field at the top of the Netscape window, you can also type URLs into that field. If you are considering taking a break, please don't. Instead, continue with the next set of steps, which show you how to add a site to your list of bookmarks, a place where you save pointers to sites in case you wish to revisit them later.

Note

Bookmarks are basically the same things as entries in a hotlist, another term you may hear.

Use A Bookmark

1. With the window to the Hewlett-Packard site still open, go to the Bookmarks menu and choose Add Bookmark.

Netscape adds the Hewlett-Packard home page to your list of bookmarks.

2. To see the bookmark in action, click on the Home button at the top of the Netscape window.

Netscape loads its default home page.

3. Drop down the Bookmarks menu and choose Access HP - Welcome to Hewlett-Packard.

Netscape takes you back to Hewlett-Packard's home page.

That's about all there is to knowing the basics of using Netscape. If you use Netscape frequently, you'll enjoy it more if you learn a few more of its features and options. I urge you to read about the section about Netscape in chapter 25, "World Wide Web," in the CD version of *Internet Starter Kit for Macintosh* and to read bits and pieces of the Handbook (choose Handbook from Netscape's Help menu).

Netscape Details

In part because of its popularity, Netscape has rather odd and somewhat irritating distribution requirements. Netscape Communications, although it would never respond to requests for explanation or clarification, has said that only sites in the **edu** domain may post Netscape for access via FTP, which eliminates all the major mirror networks because there are some sites in the **com** domain included among the mirror sites. So, finding Netscape at any site other than **ftp.netscape.com** can prove difficult, and that site is often too overloaded to serve all the folks who would like to get a copy of Netscape.

However, once you can get a copy, Netscape's wording says, "You can download a copy of Netscape Navigator for evaluation or for unlimited use in academic or not-for-profit environments. If you want to purchase Netscape Navigator and associated support for ongoing use, you can order it directly from Netscape Communications Corporation. Send email to sales@netscape.com and you'll get an automated reply with purchasing information." I'll leave it to you to interpret that wording for yourself—nowhere does Netscape give a time limit on the evaluation period. If you do decide to purchase Netscape for the support, it costs \$39, and a printed manual is another \$20. International prices vary somewhat.

<http://www.netscape.com/>

<ftp://ftp.netscape.com/netscape/mac/>

Note

For more information about other Web-related programs, scan to the end of chapter 25, "World Wide Web," in the CD version of Internet Starter Kit for Macintosh.

Helper Applications

For the most part, the various Web browsers have opted not to include the proverbial kitchen sink in their code, and they all rely on a common set of helper applications for dealing with certain types of data. There are numerous other applications that you can use as helper applications, but these are the most popular. All are available at the URL below, unless mentioned otherwise.

<ftp://ftp.tidbits.com/pub/tidbits/tisk/util/>

Note

If a Web browser cannot launch your helper applications properly, try rebuilding your desktop—if the desktop database is out-of-date, the Web browser may not be able to find the proper helper application. To rebuild your desktop, restart while holding down the \mathcal{H} and option keys until your computer asks you if you want to rebuild your desktop. Click OK and your Mac will take care of the rest for you.

Adobe Acrobat

One criticism of the HTML format used on the World Wide Web is that it doesn't provide the kind of complete control that desktop publishers are used to when laying out a page. There are a number of electronic document formats that some people use for this purpose, including Adobe Acrobat, Common Ground from No Hands Software, WordPerfect's Envoy, and Farallon's Replica. Of these, Acrobat is the most common, and Adobe and

Netscape are reportedly working to include support for Acrobat's PDF documents in future versions of Netscape. Until then, you'll have to use these programs as helper applications. I wasn't able to find a Web page for Common Ground, although the others do have Web pages.

<http://www.adobe.com/Acrobat/Acrobat0.html>

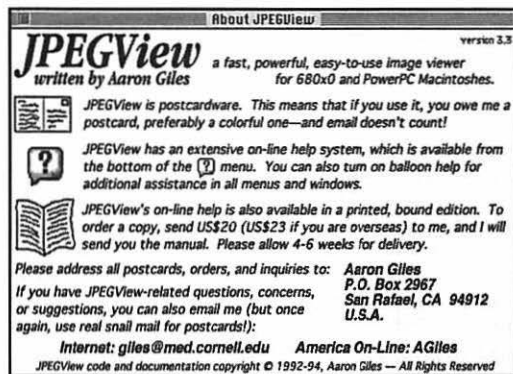
<ftp://ftp.adobe.com/pub/adobe/Acrobat/Macintosh/>

<http://wp.novell.com/envoy/envoytoc.htm>

<http://www2.farallon.com/www/www2/rep/repmac.html>

JPEGView

Although the trend among Web browsers is to display JPEG graphics along with the text and GIF graphics on a Web page, not all of them can do this yet. Those that don't send all JPEG images, and most other unsupported graphics formats, to JPEGView from Aaron Giles, which is postcardware. JPEGView does an admirable job of displaying JPEG images, and is Power Mac-native for a significant speed boost for Power Mac users. JPEGView can crop images, resize them, and convert between a number of different formats. It's a staple for other Internet programs that use helper applications as well, such as TurboGopher.



MacBinary II+

Making files available for download via a Web browser is somewhat haphazard, thanks to the way the Macintosh files can have both resources and data forks. The safest method is the standard, BinHex, but a much tighter format is MacBinary, which stores both forks of a normal Mac file together in a single file that you can upload to other types of computers. If you want to download a file in MacBinary format (usually indicated by a .bin extension), you need a helper application that can decode the MacBinary format, because Web browsers, unlike FTP clients, don't do so automatically. All that said, the program you need is Peter Lewis's free MacBinary II+ because it can work with the Web browsers to decode MacBinary files. MacBinary II+ has no interface; if a Web browser doesn't call it automatically, you must drop a MacBinary file on MacBinary II+ to have that file decoded.



MacBinary II+

MoviePlayer

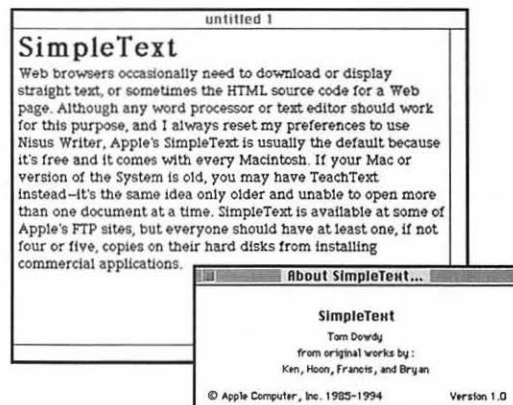
Although Sparkle, and even SimpleText, can handle QuickTime movies, Web browsers generally default to using Apple's MoviePlayer to display QuickTime movies. Of course, you must also have QuickTime installed for any of these to work. You can't download MoviePlayer, although it comes with QuickTime and on the System 7.5 CD. Getting QuickTime has become more difficult if you don't get it with your Mac or with System 7.5, so we included it on the *Internet Interactive CD* for you. Frankly, if getting MoviePlayer becomes a problem, I recommend using Sparkle, SimpleText, or one of the other free QuickTime players, like Leonard Rosenthol's Popcorn, which are readily available.



<http://quicktime.apple.com/>

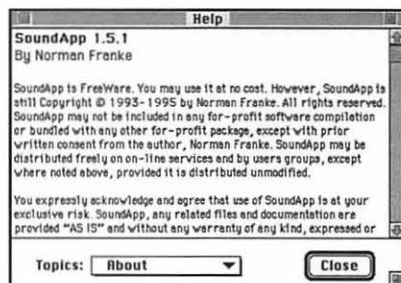
SimpleText

Web browsers occasionally need to download or display straight text, or sometimes the HTML source code for a Web page. Although any word processor or text editor should work for this purpose, and I always use Nisus Writer, Apple's SimpleText is usually the default because it's free with every Macintosh. If your Mac or version of the System is old, you may have the older and less-capable TeachText instead. SimpleText is available at some of Apple's FTP sites, and it is also included on the *Internet Interactive CD*. But everyone should have at least one, if not four or five, copies on their hard disks from installing commercial applications. SimpleText is available at some of Apple's FTP sites, but everyone should have at least one, if not four or five, copies on their hard disks from installing commercial applications. SimpleText is available at some of Apple's FTP sites, but everyone should have at least one, if not four or five, copies on their hard disks from installing commercial applications. SimpleText is available at some of Apple's FTP sites, but everyone should have at least one, if not four or five, copies on their hard disks from installing commercial applications.



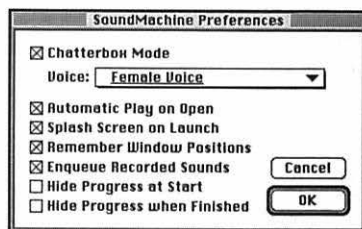
SoundApp

Although SoundMachine seems to be used as the default for most audio files, the most recent version of Norman Franke's free SoundApp claims to play even more sound formats, including one, the Windows WAVE format, that SoundMachine doesn't handle. I haven't seen many WAVE files (OK, any, but I'm not big on downloading sounds), but because they come from the Windows world, you're likely to hit some eventually. SoundApp can handle the Internet standard Sun au format, so you could use it for all of your audio needs.



SoundMachine

Rod Kennedy's free SoundMachine is the most popular application for playing the audio files that you run across on the Internet because of its support for the standard Sun au format. Though it is mostly used to play sounds, SoundMachine can also record sounds in the au format. Although it would probably drive me nuts in normal use, I rather like SoundMachine's Chatterbox Mode, where it uses Apple's PlainTalk technology to read the text of the menu items and buttons you select.



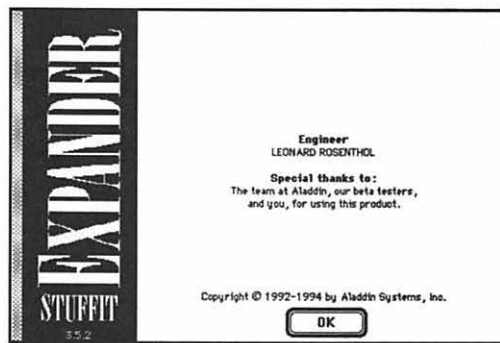
Sparkle

There are two video formats that are fairly common on the Web, QuickTime and MPEG. The most popular MPEG player for the Macintosh is Maynard Handley's free and frequently updated Sparkle. Sparkle can also display PICT files and QuickTime movies, and it contains PowerPC code for optimum movie playing performance on Power Macs. Sparkle requires System 7.5 or System 7 along with a whole slew of additional extensions, including QuickTime 1.6, Sound Manager 3.0, and the Thread Manager, among others.



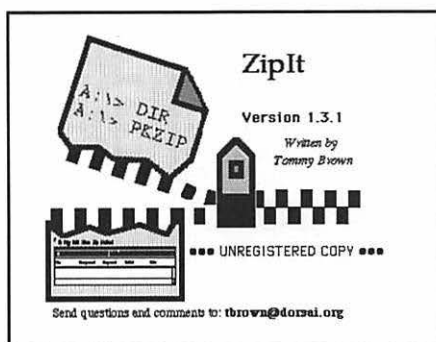
StuffIt Expander

Aladdin Systems' free StuffIt Expander is such a necessary part of your Internet toolkit that we've included it on the CD. StuffIt Expander is universally used as a helper application to debinhex files and to expand both StuffIt and Compact Pro archives, along with self-extracting archives created by either of those two programs. If you own one of Aladdin's commercial products or register their shareware DropStuff with Expander Enhancer, StuffIt Expander gains the capability to decode many other formats, include MacBinary, Unix compress, and zip files. I encourage everyone to register DropStuff with Expander Enhancer if you want to decode these additional formats—it's a great way to thank Aladdin for making the basic StuffIt Expander available for free.



ZipIt

The zip format, which is the standard in the PC world, is one of the formats that Stuffit Expander can only handle after you've registered DropStuff or purchased another Aladdin product. While you're getting around to registering DropStuff, you can try out Tommy Brown's \$15 shareware ZipIt, which also can decode (and encode) zip files. One way or another, please support these shareware efforts. Tommy Brown modeled ZipIt's interface on Compact Pro, although of course you shouldn't have to interact with the interface to a helper application all that much.



WWWrapping Up

The Internet, and specifically the Web, is where the action is, and I hope this chapter has given you the necessary information to start poking around. Everything changes so fast that it's hard to say where you should start, but I currently recommend you point your Web browser at Yahoo and either browse or search for information. If you can't find it in Yahoo, one of the search engine links in Yahoo should help, if anything can. Enjoy yourself, and remember to come up for air every now and then. The Web can be a mighty vortex.

<http://www.yahoo.com/>

In addition, I've created an ISKM Home Page (the version of MacWeb on the CD goes to it by default) that points to many of the rest of the best sites on the Internet for searching and browsing; by using these links you should be able to find anything that is actually available. These are the same resources I use, so I've given you the exact same tools to which I turn whenever I need to find something on the Web.

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

Web resources aside, this chapter brings us to the end of this book. I hope you've enjoyed it, along with its companion CD, and I hope you enjoy the Internet.

Glossary



a

address commands Small extensions for UUCP/Connect that give it additional capabilities, much as XMCDs and XFCNs give HyperCard additional capabilities.

addressing A method of identifying a resource (such as a program) or piece of information (such as a file) on a network. Methods of addressing vary considerably from network to network.

Adventure One of the earliest text adventure games written for computers. It is the forerunner of the popular Zork series from Infocom.

alias In System 7, a file that “points to” another file, folder, or disk, and may generally be used in place of the original item. In network usage, alias usually refers to a simple name, location, or command that you can use in place of a more complex name, location, or command. Aliases are commonly used for email addresses, directories, or commands.

America Online A popular commercial information service with a graphical interface.

AOL Shorthand for **America Online**. Each letter is pronounced separately.

AppleLink Apple's commercial online information service. Expensive, but graphical. Slated to go away soon.

AppleLink packages The archiving and compression format used solely by AppleLink. StuffIt Expander with Expander Enhancer can decompress AppleLink packages.

AppleTalk A local area network protocol Apple developed to connect computers and peripherals over various different types of wiring.

ARA *Apple Remote Access*. A software program from Apple Computer that allows one Mac to dial another Mac via a modem and, through AppleShare and/or Personal File Sharing, access local or network resources available to the "answering" Mac. (Common resources include shared directories, servers, and printers.) Although I don't cover the issue much in this book, you can do some neat things with ARA and MacTCP.

.ARC An older DOS archiving format.

Archie An Internet service that maintains and allows users to search a large database of materials stored on anonymous FTP sites.

archive site A site that archives files for users to retrieve, via either FTP or email.

ARPA *Advanced Research Projects Agency*. The governmental organization responsible for creating the beginnings of the Internet.

ARPAnet The proto-Internet network created by ARPA.

ASCII *American Standard Code for Information Interchange*. In the context of a file, an ASCII file is one that contains only "text" characters—numbers, letters, and standard punctuation. Although ASCII text can contain international characters available on the Mac ("upper-ASCII"), these characters are not commonly supported by Internet services such as email, Gopher, and FTP. In FTP, it's a command that tells FTP that you will be transferring text files (which is the default).

atob (pronounced "a to b") A Unix program that turns ASCII files into binary files. The **btoa** program does the reverse.

attachments Files that are linked to a specific email message, just as you might paperclip a clipping to a **snail mail** letter.

b

bandwidth Information theory used to express the amount of information that can flow through a given point at any given time. Some points have narrow bandwidth (indicating not much information can flow through at one time), and

others have high bandwidth (indicating a great deal of information can flow through at one time). This term is commonly used in reference to “wasted bandwidth,” indicating that some (or most) of the information flowing by a point is of no use to a user. This term can include overloading a site’s network connection (thus curtailing other users’ use of the lines) or including lengthy signature files in Usenet postings or discussion groups. “Wasted bandwidth” is often relative: What one person views as wasteful might be essential to someone else.

bang The exclamation point! Used to separate machine names in UUCP bang-style addressing, which isn’t all that common anymore.

baud A measure of modem speed equal to one signal per second; 300 baud equals 300 bits per second (bps). But at higher speeds one signal can contain more than one bit, so a 9600 baud modem is not a 9600 bps modem. (The terms often are incorrectly used interchangeably). See also **bps**.

BBS *Bulletin Board System*. A computer system that provides its users with files for downloading and areas for electronic discussions. Bulletin board systems usually are run by and for local users, although many now provide Internet, UUCP, or FidoNet mail.

Binary In the context of a file, any file that contains non-textual data. (Images and applications are examples of binary files.) In FTP, a command that tells FTP to transfer information as an arbitrary stream of bits rather than as a series of textual characters.

BinHex The standard Macintosh format for converting a binary file into an ASCII file that can pass through email programs. (For those of you wondering how to pronounce it, “Bin” rhymes with “tin,” and “hex” rhymes with “sex,” and the accent is on the first syllable.) See also **unicode**.

BITNET An academic large-scale computer network, primarily connecting academic institutions. BITNET is often expanded as the “Because It’s Time” Network. A friend notes, “Actually, it seems that the definitive answer to what the BIT stands for is ‘It has varied, and depends on who you asked and when.’”

BIX The online commercial information service called the *BYTE Information Exchange*, although I have never heard anyone use the full name in favor of BIX.

body The part of an email message where you type your message, as opposed to the **header** or the **signature**.

bounce What email does when it doesn’t go through.

bps *Bits per second*. The measurement of modem transmission speed. Not comparable to baud after 300 bps.

Brownian motion With apologies to Douglas Adams, the best example is indeed a really hot cup of tea. It has to do with internal movement within a hot liquid.

browser A client program that enables one to search, often somewhat randomly, through the information provided by a specific type of server. Generally used in relation to the World Wide Web.

btoa (pronounced “b to a”) A Unix program that turns binary files into ASCII files for transmission via email. The **atob** program decodes such files.

BTW Abbreviation for the expression, “*By the way.*”

C

Call For Votes What you do after discussing whether a new newsgroup should be created.

CCL *Connection Control Language*. Used in Apple Remote Access, InterSLIP, and other communications programs, CCL is a scripting language that lets you control your modem.

CERN The birthplace of the **World Wide Web**, although in real life they do high energy physics research. Located in Geneva, Switzerland. CERN doesn’t stand for anything any more, although it once was an acronym for a French name.

CFV See **Call For Votes**.

channel In IRC, an area that theoretically has a specific discussion topic. See **IRC**.

charter The document that lays out what topics a newsgroup will cover, what its name will be, and other relevant details.

chat script A simple (you hope) conversation between your Mac and your host machine that allows your Mac to log in automatically. Chat scripts usually involve a series of send and expect strings. Your host sends a login prompt; your Mac responds with your username. Your host sends a password prompt; your Mac responds with your password.

chiasmus A term from classical rhetoric that describes a situation in which you introduce subjects in the order A, B, and C, and then talk about them in the order C, B, and A.

CIM See **CompuServe Information Manager**.

CIS Stands for *CompuServe Information Service*, or simply *CompuServe*. Wags often replace the S with a \$. See **CompuServe**.

ClariNet An alternate hierarchy of newsgroups that uses the same transmission routes as Usenet, but carries commercial information from UPI and others. You, or your provider, must pay to read ClariNet news.

client The program or computer that requests information from a server computer or program. Used in terms of client/server computing. See also **server**.

clone A DOS-based computer that imitates computers made by IBM. Referred to as clones because they don't distinguish themselves enough for us to bother referring to them any other way.

CMS Short for *Conversational Monitor System*. The part of the operating system on certain IBM mainframes with which you interact. Not at all conversational.

command line Where you type commands to an operating system such as DOS or Unix. Command-line operating systems can be powerful but are often a pain to work with, especially for Macintosh users used to a graphical interface.

compress To make a file smaller by removing redundant information.

CompuServe One of the oldest and largest commercial online services. Sometimes abbreviated as **CIS**.

CompuServe Information Manager A decent graphical program for the Mac (and Windows) that puts a nice face on CompuServe. Generally abbreviated **CIM**.

connect time The amount of time you are actually connected to and using a computer. Because connect or telephone charges are based on this amount of time, you want to keep it as low as possible.

.cpt The filename extension used by Compact Pro.

CREN *Corporation for Research and Educational Networking*.

cross-posted What happens to a Usenet posting when you put several newsgroup names in the Newsgroups line. More efficient than posting multiple individual copies.

CSLIP Compressed SLIP. A type of SLIP account that uses compression to increase performance.

d

daemons Small programs in Unix that run frequently to see whether something has happened: if so, they act as they were programmed; if not, they go back to sleep.

DARPA *Defense Advanced Research Projects Agency*. Replaced ARPA and had a more military bent. Has since been renamed ARPA again. See also **ARPA**.

DEC *Digital Equipment Corporation*. Also known as *Digital*, this company produces the popular VAX line of computers and the VMS operating system.

dialing scripts In InterSLIP, the scripts that control modem dialog. See also **CCL**.

dial up To call another computer via modem. The term is often lumped together as one word except when used as a verb.

dialup A connection or line reached by modem, as in “a dialup line.”

digest A single message that contains multiple individual postings to a mailing list or newsgroup.

domain A level of hierarchy in a machine’s full nodename. For instance, **tidbits.com** is in the **com** domain, as are many other machines.

domain name server A computer that keeps track of names of other machines and their numeric IP addresses. When you refer to a machine by name, your domain name server translates that information appropriately into the numeric IP address necessary to make the connection.

domain name system The system that makes it possible for you to think in terms of names such as **penguin.tidbits.com**, whereas computers think in terms of **204.25.157.10**.

DOS An elderly operating system that is frequently helped across the street by Microsoft Windows.

download To retrieve a file from another machine, usually a host machine, to your machine.

downstream Usenet neighbors that are downstream from you get most of their news from your machine, in contrast to machines that are **upstream** from you.

dynamic addressing When your Mac gets its IP number for each session from the server to which you connect. Linked to the **Server** button in the Obtain Address part of MacTCP. Not to be confused with the **Dynamically** button, which picks an IP number from a range at random.

e

electronic mail or **email** Messages that travel through the networks rather than being committed to paper and making the arduous journey through the U.S. Postal Service.

emoticons A rather silly name for **smileys**.

Ethernet A type of local area network that is much faster than LocalTalk. Most Macs can use Ethernet by adding an Ethernet expansion card; some recent Macs come with Ethernet built-in.

.etx The filename extension for **setext** files, which are straight text files in a specific format that's easy to read online and can be decoded for even better display.

expire After a certain amount of time, Usenet postings can be set to expire, which means that they will be deleted even if they haven't been read, so that they don't waste space.

f

FAQ *Frequently Asked Question*. Lists of commonly asked questions and their answers, often posted in newsgroups to reduce the number of novice questions. Read a FAQ list before asking a question, to make sure yours isn't a frequently asked one.

Fax Slang for *facsimile*. A technology that takes paper from the sender and produces more paper that looks just like it at the recipient's end. You can use fax modems to eliminate the paper step at one end or both, but they may be less reliable than stand-alone fax machines. Email is cleaner, often cheaper, and more environmentally friendly, and the results are more useful in other programs. However, you can't easily send signatures or existing paper documents via email.

feed Shorthand for a connection to another machine that sends you mail and news. I might say, "I have a mail feed from Ed's machine."

Fidonet A network of cooperating bulletin board systems that has some links to the Internet.

filename extension A three-letter (usually) code at the end of a filename that indicates what type of file it is. Essential in non-Macintosh environments that lack icons or other methods of identifying files. Common extensions include **.txt** for text files, **.hqx** for BinHexed files, and **.sit** for Stuffit files.

fileserver or **file server** A machine that provides files via a network. Perhaps because of time spent working on BITNET, I tend to use it as a synonym for **mailserver**, or a machine that returns files that are requested via email.

file site Another name for an archive site or FTP site. A computer on which files are stored for anyone on the Internet to retrieve.

finger A method of finding out information about someone else on the Internet.

firewall A security system that not only prevents intruders from entering, but also often prevents legitimate users from getting out to the Internet from the local network. A firewall usually has a single machine that's connected to the Internet and all Internet traffic must pass through that machine.

flame war A conflagration in which lots of people jump in on different sides of an argument and start insulting each other. Fun to watch briefly, but a major waste of **bandwidth**.

flaming The act of calling into question someone's thoughts, beliefs, and parentage simply because you don't agree with them. Don't do it.

followup An article on Usenet posted in reply to another article. The subject should stay the same so that readers can tell the two articles are related.

forms In the World Wide Web, online electronic forms that you can fill in if you have a forms-capable Web browser such as MacWeb or NCSA Mosaic 2.0.

Freenet An organization whose goal is to provide free Internet access in a specific area, often by working with local schools and libraries. Ask around to see if a Freenet has sprung up in your area. The first and preeminent example is the Cleveland Freenet. Freenet also refers to the specific Freenet software, and the information services that use it.

freeware Software that you can distribute freely and use for free, but for which the author often retains the copyright, which means that you can't modify it.

FTP *File Transfer Protocol*. One of the main ways in which you retrieve, umm, well, files from other machines on the Internet.

FTPmail A method of retrieving files stored on FTP sites via email.

FYI Abbreviation for the expression, "*For your information.*"

g

gateway A machine that exists on two networks, such as the Internet and BITNET, and that can transfer mail between them.

gateway script In InterSLIP, a script that controls the login process. See also CCL.

GIF *Graphics Interchange Format*. A platform-independent file format developed by CompuServe, the GIF format is commonly used to distribute graphics

on the Internet. Mighty battles have been waged over the pronunciation of this term, and although Robin Williams notes that it's pronounced "jiff" in her book, *Jargon*, both of my glossary proofreaders flagged it as being pronounced with a hard g, as in "graphics." I surrender; pronounce it as you like.

.gif The filename extension generally given to GIF files.

GNU With apologies for the circular reference, GNU stands for *GNU's Not Unix*. Developed by Richard Stallman and the Free Software Foundation, GNU is (or will be, when finished) a high-quality version of the Unix operating system that is free of charge and freely modifiable by its users. GNU software is distributed at no cost with source code. Many GNU applications and utilities are mainstays of the Unix community.

Gopher An information retrieval system created by the University of Minnesota. In wide acceptance on the Internet, Gopher is one of the most useful resources available.

Gopherspace The collection of all available Gopher servers.

.gz An extension used by GNU's version of ZIP, called gzip.

h

hard close In MacPPP, a process that disconnects you from the Internet and prevents any programs from automatically redialing until you restart.

header The part of an email message or Usenet posting that contains information about the message, such as who it's from, when it was sent, and so on. Headers are mainly interesting when something doesn't work.

home page In the World Wide Web, the document that is accessed first after launching a Web browser.

host The large computer you connect to for your Internet access.

.hqx The filename extension used for BinHex files.

HTML *HyperText Markup Language*. The language used to mark up text files with styles and links for use with World Wide Web browsers.

HTTP *HyperText Transport Protocol*. The protocol used by the World Wide Web.

hypertext A term created by visionary Ted Nelson to describe non-linear writing in which you follow associative paths through a world of textual documents. The most common use of hypertext these days is in the links on Web pages.



i

IAB See **Internet Architecture Board**.

IBM *International Business Machines*. Many flip expansions for the acronym exist, but IBM remains one of the most powerful companies in the computer industry despite numerous problems in recent years. Developer of numerous mainframes and obtuse operating systems, some of which are still in use today. Co-developer (with Apple and Motorola) of the PowerPC chip, used in the Power Macintoshes.

IETF See **Internet Engineering Task Force**.

IMAP *Interactive Mail Access Protocol*. A new protocol for the storage and retrieval of email (much like **POP**, the Post Office Protocol). It's not in wide use yet.

IMHO Abbreviation for the expression, "*In my humble opinion*."

information agent A software program (currently only an interface to frequently updated databases) that can search numerous databases for information that interests you without your having to know what it is searching. Archie and Veronica are current examples of information agents.

internet With a lowercase *i*, it's a group of connected networks.

Internet The collection of all the connected networks in the world, although it is sometimes better called WorldNet or just the Net. More specifically, the Internet is the set of networks that communicate via TCP/IP.

Internet access provider An organization that provides Internet access for individuals or other organizations, often for a fee.

Internet Architecture Board A group of invited volunteers that manages certain aspects of the Internet, such as standards and address allocation.

Internet Engineering Task Force A volunteer organization that meets regularly to discuss problems facing the Internet.

IP *Internet Protocol*. The main protocol used on the Internet.

IP number A four-part number that uniquely identifies a machine on the Internet. For instance, my IP number for **penguin.tidbits.com** is **204.57.157.10**. People generally use the name, instead.

IRC *Internet Relay Chat*. A world-wide network of people talking to each other in real time over the Internet rather than in person.

ISOC *The Internet Society*. A membership organization that supports the Internet and is the governing body to which the IAB reports.

j

JANET *Joint Academic Network*. Great Britain's national network. In true British fashion, JANET addresses work backwards from normal Internet addresses. They work from largest domain to the smallest, as in `joe@uk.ac.canterbury.cc.trumble`. Luckily, most gateways to JANET perform the necessary translations automatically.

jargon The sometimes incomprehensible language used to talk about specialized topics. If you need help with computer jargon, check out *Jargon*, by Robin Williams, a light-hearted and detailed trip through this industry.

Jolt cola All the sugar and twice the caffeine of normal colas. First suggested as a joke by comedian George Carlin, later developed and marketed by Carlin and a food industry entrepreneur.

JPEG *Joint Photographic Experts Group*. A group that has defined a compression scheme that reduces the size of image files by up to 20 times at the cost of slightly reduced image quality.

.jpeg A filename extension used to mark JPEG-compressed images.

Jughead A searching agent for Gopher, much like Veronica, but more focused.

k

Kermit A file transfer protocol actually named after the popular Kermit the Frog. Kermit is generally slower than **XMODEM**, **YMODEM**, and the top-of-the-line **ZMODEM**.

l

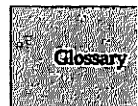
LAN See **local area network**.

leaf site A machine on Usenet that talks to only one other machine instead of passing news onto other machines.

line noise Static on a telephone line that causes trouble for modem connections.

LISTSERV A powerful program for automating mailing lists. It currently requires an IBM mainframe, but that requirement may change in the near future.

local area network Often abbreviated **LAN**. Two or more computers connected together via network cables. If you have a Macintosh connected to a LaserWriter printer (which contains a CPU), you have a rudimentary local area network.



LocalTalk The form of local area networking hardware that Apple builds into every Macintosh.

login The process by which you identify yourself to a host computer. Usually involves a userid and a password.

lurkers Not a derogatory term. People who merely read discussions online without contributing to them.

m

MacBinary A file format that combines the three parts of a Macintosh file: the data fork, resource fork, and Finder information block. No other computers understand the normal Macintosh file format, but they can transmit the MacBinary format without losing data. When you download a binary Macintosh file from another computer using the MacBinary format, your communications program automatically reassembles the file into a normal Macintosh file.

MacTCP A control panel from Apple that implements TCP on the Macintosh. MacTCP is required to use programs such as Fetch and TurboGopher.

mail bombing The act of sending hundreds or thousands of messages to someone you think deserves the punishment for transgressions against the Internet. Highly discouraged.

mailing list A list of people who all receive postings sent to the group. Mailing lists exist on all sorts of topics.

mailserver A program that provides access to files via email. See also **fileserver**.

man pages The Unix *manual pages*. You must go to the man pages to find out more about a Unix command. Accessed through use of the **man** command followed by the command whose description you want to view.

Manually A button in the MacTCP Obtain Address area. Use it if your system administrator gives you a specific IP address. Also known as **static** addressing.

MCC *Microelectronics and Computer Technology Corporation*. No, I don't know where the *T* went, either. An industry consortium that developed MacWAIS and MacWeb.

MIME *Multipurpose Internet Mail Extensions*. An Internet standard for transferring non-textual data, such as audio messages or pictures, via email.

mirror site An FTP site that contains exactly the same contents as another site. Mirror sites help distribute the load from a single popular site.

modem Stands for *modulator-demodulator*, because that's what it does, technically. In reality, a modem allows your computer to talk to another computer via the phone lines.

moderator An overworked volunteer who reads all of the submissions to a mailing list or newsgroup, to make sure they are appropriate, before posting them.

monospaced font A font whose characters are all the same width. Courier and Monaco are common monospaced fonts on the Macintosh. You generally want to use a monospaced font when reading text on the Internet.

MPEG *Motion Picture Experts Group*. More commonly, a compression format for video. Files compressed with MPEG generally have the extension **.mpeg**.

.mpeg A filename extension used to mark MPEG-compressed video.

MTU *Maximum Transmission Unit*. A number that your system administrator must give you so that you can configure SLIP.

MUD *Multi-User Dungeon*, or sometimes *Multi-User Dimension*. A text-based alternate reality where you can progress to a level at which you can modify the environment—mostly used for games, and extremely addictive.

MX record *Mail Exchange record*. An entry in a database that tells domain name servers where they should route mail so that it gets to you.

n

NCSA *National Center for Supercomputing Applications*. A group that has produced a great deal of public domain software for the scientific community. They wrote NCSA Telnet and NCSA Mosaic for the Macintosh.

net heavies Those system administrators who run large sites on the Internet. Although they don't necessarily have official posts, they wield more power than most people on the nets.

Network Information Center An organization that provides information about a network.

Network Time Protocol A protocol for transmitting the correct time around the Internet.

network time server The machine from which you set your clock using Network Time Protocol.

news Synonymous with *Usenet news*, or sometimes just *Usenet*.

newsgroup A discussion group on Usenet devoted to talking about a specific topic. Currently, approximately 9,000 newsgroups exist.

.newsrsc The file that Unix newsreaders use to keep track of which messages in which newsgroups you've read.

newsreader A program that helps you read news and provides capabilities for following or deleting threads.

NIC See **Network Information Center**.

nickname An easy-to-remember shortcut for an email address. Sometimes also called an **alias**.

NNTP *Net News Transport Protocol*. A transmission protocol for the transfer of Usenet news.

nodename The name of a machine, like **penguin.tidbits.com**.

NREN *National Research and Education Network*. The successor to the NSFNET.

NSF *National Science Foundation*. The creators of the NSFNET.

NSFNET *National Science Foundation Network*. The current high-speed network that links users with supercomputer sites around the country. Also called the interim NREN.

O

offline Actions performed when you aren't actually connected to another computer.

online Actions performed when you are connected to another computer.

p

page In the World Wide Web, the name for the basic document type.

PEP *Packetized Ensemble Protocol*. Telebit's proprietary method of increasing throughput when two of Telebit's modems connect to each other.

POP *Post Office Protocol*. A protocol for the storage and retrieval of email. Eudora uses POP.

port In software, the act of converting code so that a program runs on more than one type of computer. In networking, a number that identifies a specific

“channel” used by network services. For instance, Gopher generally uses port 70, but occasionally is set to use other ports on various machines.

post To send a message to a discussion group or list.

PPP *Point to Point Protocol*. A protocol functionally similar to **SLIP** that enables your Mac to pretend it is a full Internet machine, using only a modem and a normal telephone line.

proportionally spaced font A font whose characters vary in width, so that, for example, a *W* is wider than an *i*. Proportionally spaced fonts often work poorly when you're reading text on the Internet.

protocol A language that computers use when talking to each other.

public domain Software that you can use freely, distribute freely, and modify in any way you wish. See also **freeware** and **shareware**.

Q

QuickTime An Apple technology for time-based multiple media data. QuickTime files can include text, sound, animation, and video, among other formats. Despite being internally compressed, QuickTime movies are often huge and are hard to work with on the Internet.

quoting The act of including parts of an original message in a reply. The standard character used to set off a quotation from the rest of the text is a column of > (greater-than) characters along the left margin.

R

ranking The method by which WAIS displays found documents in order of possible utility.

relevance feedback A method WAIS uses to “find me more documents like this one.”

Request for Comments Documents containing the standards, proposed standards, and other necessary details regarding the operation of the Internet.

Request for Discussion The part of the newsgroup creation process where you propose a group and discussion begins.

RFC See **Request for Comments**.

RFD See **Request for Discussion**.

root directory The topmost directory that you can see. On the Mac, you see the root directory when you double-click on your hard disk icon.

rot13 A method of encoding possible offensive postings on Usenet so that those who don't want to be offended can avoid accidentally seeing the posting. Works by converting each letter to a number ($a = 1$, $b = 2$, and so forth), adding 13 to the number, and then converting back into letters, rendering the file unreadable without deciphering.

S

.sea The filename extension used by almost all **self-extracting archives** on the Mac.

self-extracting archive A compressed file or files encapsulated in a decompression program, so you don't need any other programs to expand the archive.

server A machine that makes services available on a network to **client** programs. A file server makes files available. A Web server makes Web pages available through the HTTP protocol.

Server A button in MacTCP's Obtain Address area that enables MacTCP to work with a dynamically addressed account.

setext *Structure-enhanced text*. A method of implicitly marking up text files to make them both easy to read online, and readable by special browser software offline.

shareware A method of software distribution in which the software may be freely distributed, and you may try it before paying. If you decide to keep and use the program, you send your payment directly to the shareware author.

signature Several lines automatically appended to your email messages, usually listing your name and email address, sometimes along with witty sayings and ASCII graphics. Keep them short, and leave out the ASCII graphics.

.sit The filename extension used by files compressed with Stuffit.

SLIP *Serial Line Internet Protocol*. Like PPP, a protocol that lets your Mac pretend it is a full Internet machine, using only a modem and a normal phone line. SLIP is older and less flexible than PPP but currently somewhat more prevalent.

smileys Collections of characters meant to totally replace body language, intonation, and complete physical presence. ; -)

SMTP *Simple Mail Transport Protocol*. The protocol used on the Internet to transfer mail. Eudora uses SMTP to send mail.

snail mail The standard name on the Internet for paper mail because email can travel across the country in seconds, whereas my birthday present from my parents once took a week.

soft close In MacPPP, the method of disconnecting from the Internet in such a way that applications can still automatically connect later on. See also **hard close**.

source In WAIS jargon, a database of information. Used interchangeably with server in the context of WAIS.

spamming The act of sending hundreds of inappropriate postings to Usenet newsgroups and mailing lists. Do it and you'll seriously regret it.

Standard File Dialog The dialog box that appears when you choose Open or Save As (and sometimes Save) from the File menu. Also known as the SFDialog.

static addressing When your Mac is assigned a permanent IP number. Most commonly used on networks that are permanently connected to the Internet. To use static addressing on the Mac, you select the Manually button in MacTCP's Obtain Address area.

system administrator The person who runs your host machine or network. Also known as the *network administrator* or just plain *administrator*. Be very nice to this person.

t

T1 A high-speed network link used on the Internet.

T3 An even higher speed network link used on the Internet.

.tar The filename extension used by files made into an archive by the Unix tar program.

TCP *Transmission Control Protocol*. It works with IP to ensure that packets travel safely on the Internet.

TCP/IP The combination of *Transmission Control Protocol* and *Internet Protocol*. The base protocols on which the Internet is founded.

Telnet Can refer to a terminal emulation protocol that lets you log in to other machines, or a program that implements this protocol on any of various platforms. On the Mac, NCSA Telnet is the standard.

terminal A piece of hardware like a **VT100** that lets you interact with a character-based operating system such as Unix.

terminal emulator Software that allows one computer, such as a Mac, to act like a dedicated terminal, such as a **VT100**.

text In terms of files, a file that contains only characters from the ASCII character set. In terms of FTP, a mode that assumes the files you will be transferring contain only ASCII characters. You set this mode in FTP with the ASCII command.

thread A group of messages in a Usenet **newsgroup** that all share the same subject and topic, so you can easily read the entire thread or delete it, depending on your specific newsreader.

TidBITS A free weekly newsletter distributed solely over computer networks. *TidBITS* focuses on the Macintosh and the world of electronic communications. I'm the editor, so I think it's neat. Send email to info@tidbits.com for subscription information.

timeout After a certain amount of idle time, some connections will disconnect, hanging up the phone in the case of a SLIP connection.

.txt The filename extension generally used for straight text files that you can read (as opposed to text files that have been encoded by BinHex or uuencode).

U

Unix An extremely popular, if utterly cryptic, operating system in wide use on computers on the Internet. Other operating systems work fine on the Internet, but Unix is probably the most common.

upload To send a file to another machine.

upstream Machines that send you most of your Usenet news are said to be upstream from you. Machines that get most of their news from you are **downstream**.

Usenet An anarchic network of sorts, composed of thousands of discussion groups on every imaginable topic.

Usenet news The news that flows through Usenet. Sometimes abbreviated *Usenet* or *news*.

userid The name you use to log in to another computer. Synonymous with **username**.

username See **userid**. They're generally the same.

.uu The filename extension generally used by uuencoded files.

uucode A file format used for transferring binary files in email, which can only reliably carry ASCII files. See also **uuencode** and **uudecode**.

UUCP *Unix to Unix CoPy*. UUCP is a small pun on the fact that the Unix copy command is cp. UUCP is a transmission protocol that carries email and news.

.uud A filename extension sometimes used by uuencoded files.

uudecode A Unix program for decoding files in the uuencode format, turning them from ASCII back into binary files. Several Macintosh programs can perform this function as well.

.uue Yet another filename extension sometimes used by uuencoded files.

uuencode A Unix program that turns binary files into ASCII files for transmission via email. Several Macintosh programs also can create uuencoded files.

V

v.34 Currently the fastest standard modem protocol, although others are due to appear soon. Although not required, almost all v.34 modems support all sorts of other protocols, including v.42 error correction and v.42bis data compression. Don't worry about the specifics; just try to match protocols with the modems you call.

Veronica An information agent that searches a database of Gopher servers to find items that interest you.

VMS DEC's main operating system for their Vax computers.

VT100 Originally, a dedicated terminal built by DEC to interface to mainframes. The VT100 became a standard for terminals, and as a result almost all terminal emulation programs can emulate the VT100. The VT100s make excellent footstools these days and will be outlived only by terminals made long ago by DataMedia that can withstand being dropped out a window without losing a connection.

W

WAIS *Wide Area Information Servers*. A set of full-text databases containing information on hundreds of topics. You can search WAIS using natural language queries and use relevance feedback to refine your search.

WAN See **wide area network**.

wide area network A group of geographically separated computers connected via dedicated lines or satellite links. The Internet enables small organizations to simulate a wide area network without the cost of one.

wildcards Special characters such as * and ? that can stand in for other characters during text searches in some programs. The * wildcard generally means “match any number of characters in this spot,” whereas the ? wildcard generally means “match any character in this spot.”

World Wide Web The newest and most ambitious of the special Internet services. World Wide Web browsers can display styled text and graphics. Often abbreviated *WWW*.

worm A program that infiltrates a computer system and copies itself many times, filling up memory and disk space and crashing the computer. The most famous worm of all time was released accidentally by Robert Morris over the Internet; it brought down whole sections of the Internet.

WWW See **World Wide Web**.

X-Z

XMODEM A common modem file transfer protocol.

YMODEM Another common modem file transfer protocol.

.Z The filename extension used by files compressed with the Unix Compress program.

.z A filename extension used by files compressed with the Unix gzip program.

.ZIP The filename extension used by files compressed into the ZIP format common on PCs.

ZMODEM The fastest and most popular modem file transfer protocol.

Index



Symbols

(hash mark), URLs, 80
/ (slash), URLs, 77
// (two slashes), URLs, 76
: (colon), URL schemes, 76, 80
; (semicolon), URLs, 79
= (equal sign), URLs, 79
? (question mark), URLs, 80
@ (at symbol), Internet addresses, 73
~ (tilde), URLs, 80

A

About Leonardo Multimedia (Internet Interactive CD), 14
Acceptable Use Policies, 59
Access HP command (Bookmark menu), 217
access providers, see *service providers*
accounts, Internet, 15, 20
Add Bookmark command (Bookmarks menu), 217
Add This Document command (Hotlist menu), 211
address commands, UUCP, 223
addresses, 223
 dynamic, 228
 email, 44, 83
 Internet sites, 67-74
 LISTSERV mailing lists, 102
 static, 239

Usenet newsgroups, 111-113
administrators, 59-60, 103, 239
Adobe Acrobat, 218-219
Advanced Research Projects (ARPAnet), 52-53
aliases, 95, 223
alternative addressing methods (Internet sites), 74
America Online, 223
Anarchie client program, 4, 48, 127, 202-206
 acquiring, 205-206
 bookmarks, 204-205
 configuring, 9, 202-205
 fees, 205
 files, retrieving, 203-204
 FTP sites, connecting to, 203-204
 location after installation, 11
Apple Computer command (Hotlist menu), 211
Apple Internet Users mailing list, 166
AppleLink, 224
AppleTalk, 224
ARA, 224
ARC filename extension, 224
archive sites, 25, 47, 224
ARPAnet (Advanced Research Projects Agency Network), 52-53, 224
Art Tour (Internet Interactive CD), 14

articles

 posting, 201
 reading, 200
ASCII (American Standard Code for Information Interchange) file formats, 146-148, 224
at symbol (@), Internet addresses, 73
atob Unix program, 224

B

Balloon Help command (Help menu), 190
bandwidth, 85, 224
bang addressing (Internet sites), 74
bangs, 225
baud rates, 225
BBSs (Bulletin Board Systems), 225
Best of Internet (Internet Interactive CD), 13
binary files, 225
 ASCII formats, 146-148
 Usenet newsgroups, downloading, 120
BinHex file format, 147-149, 225
BITNET, 55, 225
bits per second (bps), modem transmission rates, 37
BIX, 225

Blunk, Larry, MacPPP, 158, 166
body, email messages, 82-87
Bookmark menu commands, 217
bookmarks, 4
 activating, 16
 Anarchie, 204-205
 FTP sites, 127
 location after installation, 11
 Netscape, 217
Bookmarks command (File menu), 204
bounces (email messages), 90-91, 225
bps (bits per second), modem transmission rates, 37, 225
browsers, WWW, 40, 49, 143, 226
btoa Unix program, 226

C

Canter & Siegel spamming incident, 62
capital letters, email messages, 83
case sensitivity, email messages, 83
CCL, 226
CERN, 226
CFV (Call For Votes), Usenet newsgroups, 116
channels (IRCs), 131, 226
charters, Usenet newsgroups, 116, 226
chat scripts, 226
Check Mail command (File menu), 192
ClariNet, 123-124, 227
client computers, 34-35
client programs, 227
 Anarchie, 48, 127
 Archie, 48, 128-130
 FTP, 127-128
 Jughead, 49, 140
 Snatcher, 127
 Veronica, 49, 139
client/server computing, 39, 75
clones, 227
CMS (Conversational Monitor System), 227
colon (:), URLs, 76, 80
command lines, 227
commands
 Bookmark menu, 217
 File menu
 Bookmarks, 204
 Check Mail, 192
 Find command, 167

 List Bookmarks, 16
 Open Location, 216
 Open URL, 209
 Preferences, 208
 Quit, 185
 Send Queued Messages, 194
 Help menu, 190
 Hotlist menu
 Add This Document, 211
 Apple Computer, 211
 Mailbox menu, 193
 Message menu
 Delete, 194
 New Message, 192
 Reply, 194
 Navigate menu, 208
 News menu
 New Message, 201
 Next Article, 200
 Special menu, 188
commercial online services, 21, 227
Compact Pro file format, 150
composing email messages, 94-95
compression (files), 148-151, 227
CompuServe, 227
configuring
 Anarchie, 9, 202-205
 Eudora, 9, 188-194
 Internet Config, 183
 MacPPP, 6, 158-165
 MacWeb, 9-10, 207-211
 modems, 161
 Netscape, 212-217
 NewsWatcher, 9, 196-201
confirming mailing list subscriptions, 104
connect time, 227
connections, FTP sites, 127
copyrights
 Usenet newsgroups, 122
 WAIS, 136
cost, Internet connections, 57-59
cpt filename extension, 150, 227
CREN (Corporation for Research and Educational Networking), 227
cross-posting newsgroup messages, 113, 227
CSLIP, 227

D

daemons, 227
DARPA (Defense Advanced Research Projects Agency), 228
databases, WAIS, 48
Delete command (Message menu), 194
deleting email messages, 93, 193
dialing scripts, 228
dialog boxes
 Internet Preferences, 184
 MacPPP Configure Server, 161
 Settings, 9, 189
dialup (modem connections), 228
digests (mailing lists), 100-101, 228
Digital, Inc., 228
directories, WAIS, 136
disclaimers, email messages, 82
distributing newsgroup messages, 115
domain name servers, 67, 228
domains, 228
 Internet addresses, 68
 machine names, 71-72
 mid-level, 70-71
 top-level, 68-70
DOS, 228
downloading files, 120, 228
downstream computers, 240
dynamic addresses, 228

E

EarthLink Network Total Access USA, 3, 14
Echo Interval (MacPPP), 159
EINet Galaxy command (Navigate menu), 208
electronic mail, *see email*
electronic publishing, WAIS, 133-137
email, 24, 44, 228
 addresses, 44, 83
 bouncing, 225
 Eudora, 186-195
 deleting messages, 193
 reading messages, 193
 replying to messages, 193
 etiquette, 83
 file attachments, 86
 finding users, 95-97
 Knowbot Information Service, 97

- mailing lists, 97-107
- mailservers, 44
- messages, 81-91
 - body, 82-87
 - bounces, 90-91
 - capital letters, 83
 - case sensitivity, 83
 - composing, 94-95
 - deleting, 93
 - disclaimers, 82
 - emoticons/smileys, 84
 - forwarding, 94
 - headers, 87-90, 231
 - navigating, 92-93
 - reading, 91-92
 - replying, 93-94
 - signatures, 81-82
- Phone Book servers, 97
- privacy policies, 86
- programs, 91-95
- signatures, 238
- WWW access, 142
- emoticons, 84, 229
- encryption, newsgroup messages, 120
- Engst, Adam, 21-22
- equal sign (=), URLs, 79
- error messages, email transmissions, 90-91
- escape codes (URLs), 79
- Ethernet, 229
- etiquette
 - email, 83
 - FTP, 126
- etx filename extension, 229
- Eudora, 4, 186-195
 - acquiring free copy of, 195
 - configuring, 9, 188-194
 - location after installation, 11
 - messages
 - deleting, 193
 - reading, 193
 - replying to, 193
 - MIME (Multipurpose Internet Mail Extensions), 187
 - TidBITS Mailing List, subscribing to, 192-193
- F**
- FAQs (Frequently Asked Questions), 229
- faxes, 229
- feeds, 229
- Fidonet, 229
- File menu commands
 - Bookmarks, 204
 - Check Mail, 192
 - Find command, 167
 - List Bookmarks, 16
 - Open Location, 216
 - Open URL, 209
 - Preferences, 208
 - Quit, 185
 - Send Queued Messages, 194
- file servers, 229
- file sites, 47, 229
- File Transfer Protocol, *see* FTP (File Transfer Protocol)
- files
 - attaching to email messages, 86
 - binary, 225
 - compression, 227
 - extensions, 229
 - formats
 - ASCII, 146-148
 - BinHex, 147-149
 - Compact Pro, 150
 - compression, 148-151
 - MacBinary, 234
 - self-extracting archives, 150-151
 - Stuffit 3.0, 149-150
 - uucode, 241
 - FTP sites, 127-128
 - self-extracting archive, 238
 - URLs, 76
- Find command (File menu), 167
- finger program, 230
- firewalls, 230
- flaming, 34, 85, 230
- Flow Control (MacPPP), 162
- follow-up messages, newsgroups, 121-122
- fonts
 - email messages, viewing, 92
 - monospaced, 235
 - proportionally spaced, 237
- formats, files
 - ASCII, 146-148
 - BinHex, 147-149
 - Compact Pro, 150
 - compression, 148-151
 - MacBinary, 234
 - self-extracting archives, 150-151
 - Stuffit 3.0, 149-150
 - uucode, 241
- forms (WWW), 230
- forwarding email messages, 94
- Freenet, 230
- freeware, 25, 230
- FTP (File Transfer Protocol), 25, 47-48, 125-130, 230
- Anarchie client program, 202-206
- Archie client program, 128-130
- client programs, 127-128
- Kermit, 233
- sites
 - bookmarks, 127
 - Compact Pro, 150
 - connecting to, 16, 127
 - files, 127-128
 - mirror sites, 47, 234
 - navigating, 127
 - Stuffit, 150
 - XMODEM, 242
 - YMODEM, 242
 - ZMODEM, 242
- FTPmail, 230
- future Internet development, 60-63
- FYI (For Your Information), 230
- G**
- gateway scripts, 230
- gateways, 53-54, 230
- geography, Internet, 31-32
- GIF files, 230
- GNU operating system, 231
- Gopher, 25, 48-49, 137-140, 231
 - Jughead, 140
 - sites, 41
 - URLs, 79
 - Veronica, 139
- Gopherspace, 231
- government regulation, 63
- Green Card debacle, 61
- H**
- hard close (MacPPP), 231
- hardware handshaking cables, 173
- hardware Internet connectivity, 34
- hash marks (#), URLs, 80
- headers, email messages, 87-90, 231
- Help menu commands, Balloon Help, 190
- home pages (WWW), 141, 231
- host computers, 34-35, 42-43, 231
 - see also* servers
- Hotlist menu commands, 211
- hqx filename extension, 147, 231
- HTML (HyperText Markup Language), 141, 212, 231

HTTP (HyperText Transfer Protocol), 77, 140, 231
HyperText, 50, 141, 231

I

IAB (Internet Architecture Board), 59, 232
IBM (International Business Machines), 232
Idle Timeout (MacPPP), 159
IETF (Internet Engineering Task Force), 59, 232
IMAP (Interactive Mail Access Protocol), 232
In command (Mailbox menu), 193
information agents, 232
information resources, 41-50
inline graphics, Web pages, 141
installing
 ISKM Installer, 5-10
 MacPPP, 158-165
Integrated Services Digital Network, see *ISDN*
Internet, 19-22
 accounts, 14-15, 20
 connecting to, 3-4
 development, 51-63
 popularity, 26
 resources, 27-28
 size, 29-32
 trends, 22
Internet Config, 4, 181-186
 configuring, 183
 location after installation, 11
 online support, 186
 preferences, 182
Internet Interactive CD, 12-14
 contents, 12-14
 MacTCP, 155-179
 system requirements, 20-21
Internet Preferences dialog box, 184
Internet Starter Kit for Macintosh, Second Edition, 30
internetworks, 35, 232
InterSLIP, gateway scripts, 230
InterText magazine, 25
IP (Internet Protocol), 53, 232
IRC (Internet Relay Chats), 131, 226, 232
ISDN (Integrated Services Digital Network) connect, 36
ISKM Home Page, 16-17
ISKM Installer
 file location, 10-12
 installing, 5-10

Internet programs, 4-12
MacPPP, installing and configuring, 158-165
MacTCP, 156
ISOC (The Internet Society), 232

J-K

JANET (Joint Academic Network), 233
JPEG files, 233
JPEGView, 219
Jughead client program, 49, 140, 233
Kermit (FTP), 233
Knowbot Information Service (email), 97
Knowledge Navigator, 25

L

LANs (local area networks), 35, 233
leaf sites, 233
Leonardo Internet accounts, 4
Lewis, Peter, MacTCP Watcher, 169
Library of Congress Gopher site, 41
line noise (modems), 38, 233
List Bookmarks command (File menu), 16
listing mailing list subscribers, 105
ListProcessor mailing lists, 106
LISTSERV mailing lists, 98, 233
 addresses, 102
 manager programs, 101-107
 subscribing, 102
 switching between message/digest formats, 106
Little Mac Book, Robin Williams, 21
local area networks, see *LANs*
LocalTalk, 234
login, 234
lurking, 234
Lynx browser, 40

M

MacBinary file format, 219, 234
machine names, Internet address domains, 71-72
Macintosh file sites, 47
MacPPP, 4, 157-166
 advantages, 165-166
 Blunk, Larry, 166
 configuring, 6
 Echo Interval feature, 159

Flow Control feature, 162
 hard close, 231
Idle Timeout feature, 159
installing and configuring, 158-165
 location after installation, 11
 modems, configuring, 161
 problems
 reporting, 167-169
 troubleshooting, 175-179
 soft close, 239
Terminal Window
 checkbox, 160
 troubleshooting, 166-179
MacPPP Configure Server dialog box, 161
MacTCP, 4, 155-179, 234
 domain servers, troubleshooting problems, 171-175
 location after installation, 10
 problems, reporting, 167-169
 programs, 181-222
 troubleshooting, 166-179
MacTCP Watcher, 169
MacWeb, 4, 206-211
 acquiring free copy of, 211
 configuring, 9-10, 207-211
 hotlists, 211
 ISKM Home Page, connecting to, 16-17
 location after installation, 11
 World Wide Web, navigating, 208-209
mail bombing, 62, 234
mail, see *email*
Mailbox menu commands, 193
mailing lists, 44, 97-107, 234
 administrators, 103
 digests, 100-101
 listing subscribers, 105
 ListProcessor, 106
 Majordomo, 107
 manager programs, 101-107
 messages, 100-101
 moderated, 99-100
 moderators, 235
 posting messages, 103
 signing off, 105
 subscribing, 102, 104
 unmoderated, 99-100
 unsubscribe, 106
 Usenet, 45
 vs. newsgroups, 98, 110
mailservers, 44, 234

- Majordomo mailing lists, 98, 107
- Managing Internet Information Services*, 35
- manual pages (Unix), 234
- marking newsgroup messages, 119
- MCC (Microelectronics and Computer Technology Corporation), 234
- Message menu commands
 - Delete, 194
 - New Message, 192
 - Reply, 194
- mid-level domains, Internet addresses, 70-71
- Milnet (Military Network), 53
- MIME (Multipurpose Internet Mail Extensions), 187, 234
- mirror sites (FTP), 47, 234
- modems, 36-39, 235
 - baud rates, 225
 - configuring, 161
 - dialup, 228
 - v.34 protocol, 241
- moderated mailing lists, 99-100
- moderators (mailing lists/newsgroups), 59-60, 235
- monospaced fonts, 235
- MoviePlayer, 220
- MPEG files, 235
- MTU (Maximum Transmission Unit), SLIP, 235
- MUDs (Multi-User Dimensions/Dungeons), 132-133, 235
- MX (Mail Exchange) records, 235
- N
- National Research and Education Network (NREN), 56
- National Science Foundation Network (NSFNET), 55-56
- Navigate menu commands, 208
- navigating
 - email messages, 92-93
 - FTP sites, 127
 - Usenet newsgroups, 118-120
 - World Wide Web, Netscape, 214-215
- NCSA (National Center for Supercomputing Applications), 235
- Netscape, 212-218
 - acquiring copy of, 217-218
 - bookmarks, 217
 - configuring, 212-217
 - HTML (HyperText Markup Language), 212
 - World Wide Web, navigating, 214-217
- Network Information Center, 235
- Network Time Protocol, 235
- network time servers, 235
- networks
 - administrators, 59
 - Internet connectivity, 35-36
 - Internet links, 34
- New Message command (Message menu), 192
- New Message command (News menu), 201
- News menu commands
 - New Message, 201
 - Next Article, 200
- newsgroups, 24, 45, 236 (ClariNet), 123-124
 - addresses, 111-113
 - articles
 - posting, 201
 - reading, 200
 - binary files, downloading, 120
 - categories, 111
 - CFV, 116
 - charters, 116, 226
 - connection problem consultation, 168
 - copyright infringements, 122
 - creating, 115-117
 - cross-posting messages, 227
 - flame wars, 34
 - follow-up messages, 121-122
 - marking messages, 119
 - message expiration, 45
 - moderators, 235
 - navigating, 118-120
 - posting messages, 110, 113-115, 122-123
 - reading, 117-118
 - replying to messages, 114, 121
 - RFDs, 116
 - rot13, 120
 - subscribing to, 117, 198-199
 - threads, 110, 118, 240
 - vs. mailing lists, 110
- newsrsrc filename extension, 236
- newsreaders, 114, 117-118, 236
- NewsWatcher, 195-202
 - acquiring free copy of, 202
 - articles
 - posting, 201
 - reading, 200
 - configuring, 9, 196-201
 - subscription lists, creating, 198-199
- Next Article command (News menu), 200
- nicknames, 236
- NNTP (Net News Transport Protocol), 236
- nodenames, 236
- NREN (National Research and Education Network), 56, 236
- NSF (National Science Foundation), 236
- NSFNET (National Science Foundation Network), 55-56
- number sign (#), URLs, 80
- O
- offline, 236
- online, 236
- Open Location command (File menu), 216
- Open URL command (File menu), 209
- operating systems
 - DOS, 228
 - GNU, 231
 - VMS, 241
- O'Reilly & Associates, 35
- P
- PEP (Packetized Ensemble Protocol), 236
- Phone Book servers (email), 97
- phone lines, Internet connectivity, 36-39
- POP (Post Office Protocol), 236
- ports, 42, 236
- posting
 - articles, NewsWatcher, 201
 - mailing list messages, 103
 - newsgroup messages, 110, 113-115, 122-123, 201
- PPP (Point to Point Protocol), 157, 237
- preferences, Internet Config, 182
- Preferences command (File menu), 208
- programs
 - Adobe Acrobat, 218-219
 - Anarchie, 4, 202-206
 - Bookmarks, 4, 16

- client/server computing, 39-40
- email, 91-95
- Eudora, 4, 186-195
- Finger, 230
- information agents, 232
- Internet Config, 4, 181-186
- ISKM Installer, 4-12
- JPEGView, 219
- MacBinary, 219
- MacPPP 2.0.1, 4
- MacTCP, 4, 156-179
- MacWeb, 4, 206-211
- MoviePlayer, 220
- Netscape, 212-218
- NewsWatcher, 9, 195-202
- QuickTime 2.0, 4
- SimpleText, 220
- SoundApp, 220
- SoundMachine, 221
- Sparkle, 221
- Stuffit Expander, 4, 221
- uudecode, 241
- uuencode, 241
- worm, 242
- Ziplt, 222
- proportionally spaced fonts, 237
- protocols, 237
 - FTP, 47-48, 125-130, 230
 - HTTP, 140, 231
 - IMAP, 232
 - IP, 53, 232
 - Kermit (FTP), 233
 - Network Time Protocol, 235
 - NNTP, 236
 - PEP, 236
 - POP, 236
 - PPP, 237
 - SLIP, 238
 - SMTP, 239
 - TCP, 239
 - TCP/IP, 35, 239
 - Telnet, 46, 130-131
 - URL schemes, 76
 - UUCP, 36, 241
 - v.34 (modems), 241
 - WAIS, 133-137
 - XMODEM (FTP), 242
 - YMODEM (FTP), 242
 - ZMODEM (FTP), 242
- public domain software, 237
- punctuation
 - Internet addresses, 73-74
 - Usenet newsgroup addresses, 111

Q-R

- Qualcomm mailing list, subscribing to, 195
- question marks (?), URLs, 80
- QuickTime, 4, 11, 237
- Quit command (File menu), 185
- quoting (messages), 237
- ranking (WAIS), 134, 237
- Read the Book (*Internet Interactive CD*), 13
- reading
 - articles, NewsWatcher, 200
 - email messages, 91-92, 193
 - Usenet newsgroups, 117-118, 200
- relevance feedback, WAIS, 134, 237
- Reply command (Message menu), 194
- replying
 - email messages, 93-94
 - Usenet newsgroup messages, 114, 121, 193
- RFCs (Requests for Comment), 43
- RFDs (Requests for Discussion), 116
- root directories, 238
- rot13 cryptosystem, 120, 238

S

- Scully, John, Knowledge Navigator, 25
- sea filename extension, 151, 238
- searching agents, WWW site, 142
- self-extracting archive file format, 150-151, 238
- semicolons (;), URLs, 79
- Send Queued Messages command (File menu), 194
- servers, 42-43, 229, 238
 - Archie, 129-130
 - Gopher, 138
 - ports, 42
 - URL schemes, 75
 - Veronica, 139
 - WAIS, 133-137
 - see also host computers
- service providers, 14-15, 20, 57-59
 - BIX, 225
 - commercial, CompuServe, 227
- setext (Structure-enhanced text), 238

- Settings command (Special menu), 188
- Settings dialog box, 9, 189
- SF (Standard File) dialog box, 239
- shareware, 25, 238
- signatures, email messages, 81-82, 238
- signing off mailing lists, 105
- SimpleText, 220
- sit filename extension, 149, 238
- sites
 - archive, 224
 - file, 229
 - FTP
 - Compact Pro, 150
 - connections, 127
 - navigating, 127
 - Stuffit, 150
 - Gopher, Library of Congress, 41
 - leaf, 233
 - mirror (FTP), 234
 - Telnet, Weather Underground, 46
 - WWW
 - Internet users, 30
 - searching agents, 142
 - Wired magazine, 132
- slashes (/), URLs, 77
- SLIP, 238
 - MTU (Maximum Transmission Unit), 235
 - timeouts, 240
- smileys, 84, 238
- SMTP, 239
- snail mail, 239
- Snatcher client program, 127
- soft close (MacPPP), 239
- software, see *programs*
- SoundApp, 220
- SoundMachine, 221
- spambots, 62
- spamming, 61, 239
- Sparkle video player, 221
- Special menu commands, 188
- static addresses, 239
- Step-by-Step Internet (*Internet Interactive CD*), 13
- stubs, compressed files, 150
- Stuffit Expander 3.0, 4, 11, 149-150, 221
- subscribing to
 - LISTSERV mailing lists, 102
 - newsgroups, 117, 198-199
- system requirements, *Internet Interactive CD*, 20-21

T

T1 network links, 239
 T3 network links, 239
 Talk to Adam (*Internet Interactive* CD), 13
 TCP, 239
 TCP/IP (Transmission Control Protocol/Internet Protocol), 35, 125-143, 156, 239
 telephone lines, Internet connectivity, 36-39
 Telnet, 46, 130-131, 239
 terminal emulation programs, 240
 Terminal Window checkbox (MacPPP), 160
 terminals, 240-241
 text, 240
 ASCII characters, 224
 setext, 238
 threads, newsgroups, 110, 118, 240
 TidBITS, 240
 email address, 44
 Internet sites, 61
 mailing list, subscribing to, 192-193
 newsletter, 25
 tildes (~), URLs, 80
 timeouts (SLIP), 240
 top-level domains, Internet addresses, 68-70
 Transmission Control Protocol/Internet Protocol, see *TCP/IP*
 troubleshooting modems, 38
 TurboGopher, 139
 two slashes (/ /), URL schemes, 76

U

Uniform Resource Locators, see *URLs*
 University of Michigan
 Weather Underground
 Telnet, 46
 Unix, 240
 Internet servers, 35
 manual pages, 234
 Unix to Unix CoPy, see *UUCP*
 unmoderated mailing lists, 99-100
 unsubscribing to mailing lists, 106
 uploading files, 240
 upstream computers, 240
 URLs (Uniform Resource Locators), 67, 75-80

Gopher, 79
 using, 78-79
 Usenet, 54
 articles
 posting, 201
 reading, 200
 mailing lists, 45
 newsgroups, 24, 45, 240
 addresses, 111-113
 age/gender survey, 109
 binary files
 categories, 111
 CFV, 116
 charters, 116, 226
 connection problem
 consultation, 168
 copyright infringements, 122
 creating, 115-117
 cross-posting messages, 227
 downloading, 120
 follow-up messages, 121-122
 marking messages, 119
 moderators, 235
 navigating, 118-120
 posting messages, 110, 113-115, 122-123
 reading, 117-118
 replying to messages, 114, 121
 RFDs, 116
 rot13, 120
 subscribing, 117, 198-199
 threads, 110, 118, 240
 see also newsgroups
 NewsWatcher, 195-202
 userids, 72-73, 240
 usernames, Internet address domains, 72-73
 utilities
 Compact Pro, 150
 self-extracting archives, 150-151
 StuffIt, 149-150
 uu filename extension, 241
 uucode file format, 241
 UUCP (Unix to Unix CoPy) protocol, 36, 223, 241
 uuencode program, 241
 uue filename extension, 241
 uuencode program, 241

V-W

v.34 modem protocol, 241
 Veronica client program, 49, 139, 241
 viewing FTP files, 127
 virtual ports, servers, 42
 VMS operating system, 241
 VT100 terminals, 241
 WAIS (Wide Area Information Server), 48, 133-137, 241
 copyright infringements, 136
 Directory of Servers, 136
 ranking, 134, 237
 relevance feedback, 134, 237
 WANs (wide area networks), 242
 Weather Underground Telnet site, 46
 Web pages, 141
 Welty, Eudora, Why I Live at the Post Office, 187
 wildcards, 242
 Wired magazine, WWW site, 132
 World Wide Web, see *WWW*
 worm programs, 242
 WWW (World Wide Web), 49-50, 140-143, 242
 browsers, 40, 143, 226
 email access, 142
 forms, 142, 230
 helper applications, 218-222
 home pages, 141
 ISKM Home Page, connecting to, 16-17
 MacWeb, 206-211
 Netscape, 212-218
 sites, estimated users, 30-31
 searching agents, 142
 Wired Magazine
 URLs (Uniform Resource Locators), 79

X-Z

XMODEM (FTP), 242

YMODEM (FTP), 242

Z filename extension, 242

ZIP filename extension, 242

Ziplt, 222

ZMODEM (FTP), 242

MacWeb and winWeb

Your customers, employees, and partners need information -- information that helps them make decisions, evaluate problems, and accomplish tasks. The longer it takes them to find the information, the more it costs.

Finding information with TradeWave client products is as easy as asking for it. And coupled with TradeWave secure servers, you control the information returned.

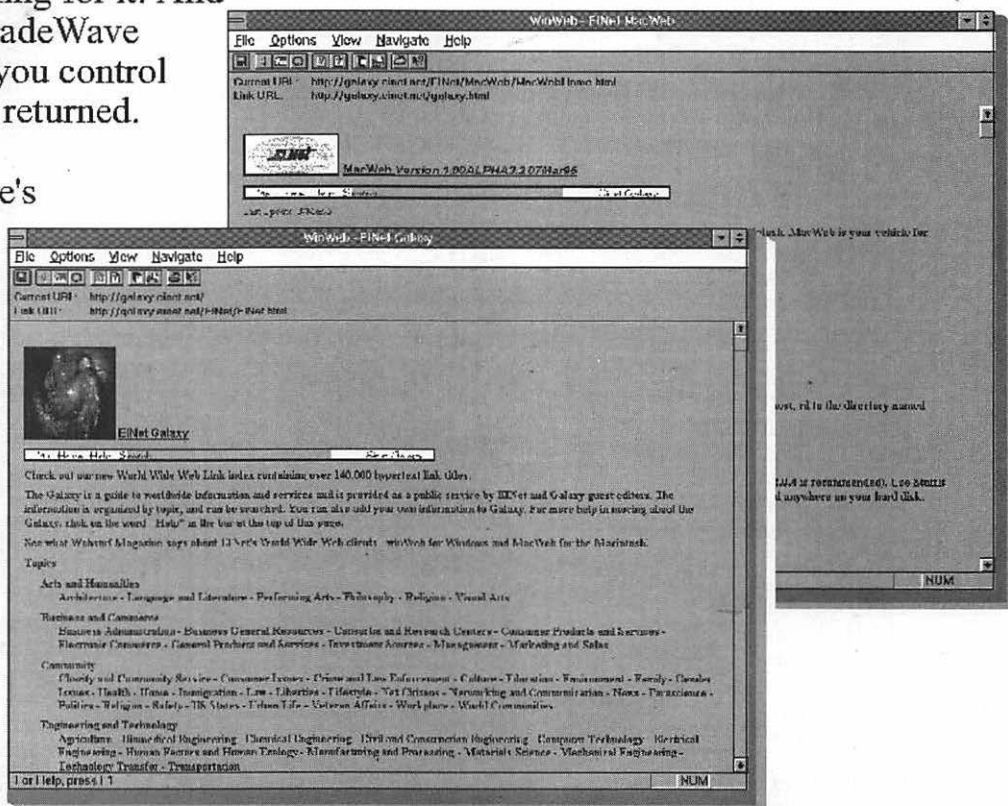
With TradeWave's World-Wide Web (WWW) clients, users don't need to know where information is located.

Users choose what they want to look at. MacWeb and winWeb locate and retrieve it.

Users don't need to settle for what they can find -- instead they can find what they need.

MacWeb and winWeb fully exploit the Web's hypermedia technology while maintaining the look and feel of the operating platform.

Power and Simplicity



TradeWave Corporation

**3636 Executive Center Drive
Suite 100**

Austin, Texas 78731

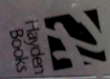
Sales and Marketing Department

512-433-5300

512-433-5303 Fax



INTERNET
INTERACTIVE



© 1995 HAYDEN BOOKS • ISBN 1-56830-220-4
LEONARDO MULTIMEDIA AND WILAND PRODUCTIONS
All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or by any information storage and retrieval system, without permission in writing from the publisher.



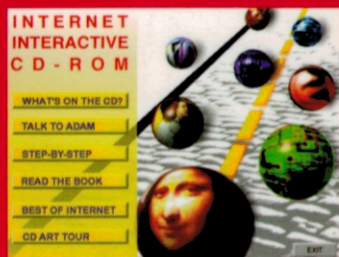
YOUR NO SWEAT, STEP-BY-STEP, TICKET TO THE INTERNET!



Embark on a magical mystery voyage of the Internet with *Simply Amazing Internet for Macintosh*® as your magic carpet! The incredible frontier of cyberspace—once thought of as no man's land—is now the

hottest, most talked about technology in the world. Simply strap yourself into your chair, slide in your disc, and in a few quick steps you will be using the Net with the most exciting and stimulating graphics anywhere.

With *Simply Amazing Internet for Macintosh* you get easy-to-understand, step-by-step instructions, a CD-ROM that not only



gets you on the Internet, but also gives you a **multimedia tour**—letting you hear and see what the Net is all about.

REQUIREMENTS:

System 7.0 or higher
2400 bps modem
(9600 bps recommended)
4 MB of RAM (8 MB for Power Macintosh)
CD-ROM drive



Cover image and back cover screen shots licensed by Bughouse Graphics



QuickTime is a trademark of Apple Computer, Inc.

Hayden Books

Category: Communications/Online

User Level:

New Casual Accomplished Expert

Covers: The Internet



CD-ROM Includes:

- All the software you need to get on the Net
- More than 100 freeware and shareware Internet programs
- An Internet tutorial featuring step-by-step, point-and-click examples to get you on the Net
- Interviews with guru Adam Engst, author of the bestselling *Internet Starter Kit for Macintosh, 3rd Edition*
- Some of the hottest sites on the Internet for you to browse—all you have to do is point & click

\$39.95 USA / \$53.99 CAN/\$37.50 UK (inc of vat)

