MRCRR

FOX

HTML WEB PUBLISHER'S CONSTRUCTION KIT

QA 76
.76
.H94 F69
1995

THE WAITE GROUP®

COVERS NEWEST HTML 3 SPECIFICATIONS

HTML WEB PUBLISHER'S CONSTRUCTION KIT

PUBLISHING YOUR OWN HTML PAGES ON THE INTERNET

CD-ROM comes with:
NetManage Chameleon • SlipKnot
HoTMetaL • HTML Assistant
WinHTTPd • LView • MPEG Play
Wham • CGI scripts for
AppleScript, Unix, and Windows
NCSA's HTTPd for Unix • and more!

WAITE GROUP PRESS™

DAVID FOX
TROY DOWNING

TX 4-100-720
TX0041100720

HTML WEB PUBLISHER'S CONSTRUCTION KIT

LIBRARY OF CONGRESS
SEP 19 1995
COPYRIGHT OFFICE
9

WAITE GROUP PRESS™
Corte Madera, CA

DAVID FOX
TROY DOWNING

QA76
.76
.H94F69
1995
MRCPR

Publisher • **Mitchell Waite**
Editor-in-Chief • **Charles Drucker**
Editorial Director • **John Crudo**
Managing Editor • **Dan Scherf**
Copy Editor • **Hilary Powers**
Technical Editor • **Miko Matsumura**
Production Director • **Julianne Ososke**
Production Manager • **Cecile Kaufman**
Production Coordinator • **Ingrid Owen**
Cover Design • **Sestina Quarequio**
Designer • **Karen Johnston**
Illustrations • **Kristin Peterson**
Production • **Michele Cuneo**

© 1995 by The Waite Group, Inc.®
Published by Waite Group Press™, 200 Tamal Plaza, Corte Madera, CA 94925.

Waite Group Press is distributed to bookstores and book wholesalers by Publishers Group West, Box 8843, Emeryville, CA 94662, 1-800-788-3123 (in California 1-510-658-3453).

All rights reserved. No part of this manual shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, desktop publishing, recording, or otherwise, without permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. 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 contained herein.

All terms mentioned in this book that are known to be registered trademarks, trademarks, or service marks are listed below. In addition, terms suspected of being trademarks, registered trademarks, or service marks have been appropriately capitalized. Waite Group Press cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any registered trademark, trademark, or service mark.

The Waite Group is a registered trademark of The Waite Group, Inc.
Waite Group Press and The Waite Group logo are trademarks of The Waite Group, Inc.
All other product names are trademarks, registered trademarks, or service marks of their respective owners.

Printed in the United States of America
95 96 97 98 • 10 9 8 7 6 5 4 3 2 1

Fox, David
HTML Web publisher's construction kit / David Fox, Troy Downing
p. cm.
Includes index.
ISBN: 1-57169-018-2
1. Hypertext systems. 2. HTML (Document markup language)
I. Downing, Troy. II. Title.
QA76.76.H94F69 1995
005.75--dc20

95-19286
CIP

DEDICATION

To Amy. Yup, I said Amy.
— *David Fox*

I would like to dedicate this to my wife Laura for putting up with my late nights at work, and to my daughter Morgan—may this help pay your college tuition. —*Troy Downing*

Message from the **Publisher**

Welcome to Our Nervous System

Some people say that the World Wide Web is a graphical extension of the information superhighway, just a network of humans and machines sending each other long lists of the equivalent of digital junk mail.

I think it is much more than that. To me the Web is nothing less than the nervous system of the entire planet—not just a collection of computer brains connected together, but more like a billion silicon neurons entangled and recirculating electro-chemical signals of information and data, each contributing to the birth of another CPU and another Web site.

Think of each person's hard disk connected at once to every other hard disk on earth, driven by human navigators searching like Columbus for the New World. Seen this way the Web is more of a super entity, a growing, living thing, controlled by the universal human will to expand, to be more. Yet unlike a purposeful business plan with rigid rules, the Web expands in a nonlinear, unpredictable, creative way that echoes natural evolution.

We created our Web site not just to extend the reach of our computer book products but to be part of this synaptic neural network, to experience, like a nerve in the body, the flow of ideas and then to pass those ideas up the food chain of the mind. Your mind. Even more, we wanted to pump some of our own creative juices into this rich wine of technology.

Taste our Digital Wine

And so we ask you to taste our wine by visiting the body of our business. Begin by understanding the metaphor we have created for our Web site—a universal learning center, situated in outer space in the form of a space station. A place where you can journey to study any topic from the convenience of your own screen. Right now we are focusing on computer topics, but the stars are the limit on the Web.

If you are interested in discussing this Web site, or finding out more about the Waite Group, please send me email with your comments and I will be happy to respond. Being a programmer myself, I love to talk about technology and find out what our readers are looking for.

Sincerely,

Mitchell Waite

Mitchell Waite, C.E.O. and Publisher

200 Tamal Plaza
Corte Madera CA 94925
415 924 2575
415 924 2576 fax

Internet email:
mwaite@waite.com

CompuServe email:

75146,3515

Website:
http://www.waite.com/waite

Creating the Highest Quality Computer Books in the Industry

Waite Group Press
Waite Group New Media

Come Visit

WAITE.COM

Waite Group Press World Wide Web Site

Now find all the latest information on Waite Group books at our new Web site, **http://www.waite.com/waite**. You'll find an online catalog where you can examine and order any title, review upcoming books, and send email to our authors and editors. Our ftp site has all you need to update your book: the latest program listings, errata sheets, most recent versions of Fractint, POV Ray, Polyray, DMorph, and all the programs featured in our books. So download, talk to us, ask questions, on **http://www.waite.com/waite**.

The New Arrivals Room has all our new books listed by month. Just click for a description, Index, Table of Contents, and links to authors.

The Backlist Room has all our books listed alphabetically.

The People Room is where you'll interact with Waite Group employees

Links to Cyberspace get you in touch with other computer book publishers and other interesting Web sites.

The FTP site contains all program listings, errata sheets, etc.

The Order Room is where you can order any of our books online.

The Subject Room contains typical book pages which show description, Index, Table of Contents, and links to authors.

World Wide Web:

http://www.waite.com/waite
Gopher: gopher.waite.com
FTP: ftp.waite.com

COME SURF OUR TURF—THE WAITE GROUP WEB

TM

ABOUT THE AUTHORS

David Fox lives in New York City, where he writes novels, screenplays, articles, and technical stuff, depending on how broke he is. Currently he's keeping pretty darn busy programming for a major multimedia company. He's the author of several books, including The Waite Group's *Love Bytes: The Online Dating Handbook*. Reach him at http://found.cs.nyu.edu/dfox.

Troy Downing is a Research Scientist, Programmer, System Administrator, and Web Administrator at New York University's Media Research Lab. Troy is a primary force in WWW development at the Lab and has helped set up WWW services for NYU's Computer Science Department and the Center for Advanced Technology, as well as the Media Research Lab. When not sitting in front of a computer terminal, you will find Troy playing with his daughter, racing mountain bikes, or perfecting his latest batch of homebrew.

Troy can be found at: http://found.cs.nyu.edu/downing.

CONTENTS

INTRODUCTION XXIV
THE CHAMELEON SAMPLER XXVII

PART I CONNECTING TO THE WEB 1
1 CATCHING THE INTERNET IN A WEB 1
2 WEB BROWSERS 39
3 LYNX 53
4 SLIPKNOT 79
5 NCSA MOSAIC 103
6 THE NETSCAPE NAVIGATOR 127
7 A BRIEF LOOK AT OTHER BROWSERS 161

PART II CREATING WEB PAGES 173
8 WHAT CAN I DO?. 173
9 THE GAME PLAN 223
10 THE HYPERTEXT MARKUP LANGUAGE 243
11 TEXT 259
12 GRAPHICS 319
13 SOUND 373
14 INTERACTIVITY 387
15 CGI SCRIPTS 417
16 OTHER WEB RESOURCES 467
17 CONVERTING, TRANSLATING, OR CHEATING 485
18 HTML ASSISTANT 505
19 HoTMetaL 525

PART III WEAVING A WEB OF YOUR OWN 555
20 WHERE TO PLACE YOUR HTML DOCUMENTS 555
21 STARTING YOUR OWN WEB SITE 579
A SLIP INTERNET PROVIDERS 631
B OTHER SOFTWARE SOURCES 637
C WEB HELP RESOURCES 647
D ABOUT THE CD 653
INDEX 657

TABLE OF CONTENTS

PART I CONNECTING TO THE WEB I

I CATCHING THE INTERNET IN A WEB I

HOW THE WEB WAS WOVEN 4
CERN: The Concept 5
NCSA: The Tool 5
The New Fabric 5
THE WEB'S EVOLUTIONARY EDGE 6
WORLD WIDE WEB LAYOUT 7
The Server 8
The Client Browser 8
NAVIGATING WEB PAGES 8
Hypermedia Links 9
Search Indexes 10
THE HTTP PROTOCOL 12
URLS, URLS, URLS 13
The Composition of a URL 14
Services 14
ACCESSING THE WEB 18
What Computer Do I Need? 18
What About the Modem? 19
What Type of Internet Account Do I Need? 19
SLIP ACCOUNTS 20
TCP/IP 21
Installing and Configuring Trumpet Winsock 22
Manual Login 25
Login Scripting 26
Logging Out 28
UNIX SHELL WEB BROWSERS 29
Text: Lynx or WWW 29
Graphics: SlipKnot 30
THE INTERNET ADAPTER 30
Determining Your Unix Platform 30
Getting Your License Code 32
Getting the Software 33
Installing The Internet Adapter 34
Configuring Winsock for The Internet Adapter 34
Exiting TIA 36
SHOOTING DOWN TROUBLES 36

Your Modem Won't Respond 36
The "DNS Lookup Failed" Message 37
The "Unable to Resolve Host Name" Message 37
WHAT NOW? 38

2 WEB BROWSERS 39
WHAT TYPE OF SOFTWARE IS A BROWSER? 41
WHICH BROWSER SHOULD I USE? 43
HELPFUL HANDS 44
Viewers 44
External Applications 44
The Talking MIME 45
TELNETING TO A PUBLIC BROWSER 47
WWW VIA E-MAIL 49
BROWSING WITHOUT THE INTERNET ALTOGETHER 50
Stand-Alone Browsers 50
Minimizing Winsock 50
Nullsock 50
WHAT NOW? 51

3 LYNX 53
LESSON #1: THE LYNX'S MEOW 57
LESSON #2: NAVIGATING 58
Following Lynx Links 59
Link Info 59
Loading a Specific URL 59
Halting a Load 60
Exiting Lynx 60
LESSON #3: THE HISTORY LIST 60
LESSON #4: SCREEN-WASHING 61
LESSON #5: PRINTING A PAGE 61
Saving 62
Mailing 62
Printing 62
LESSON #6: GETTING TO THE SOURCE 63
Saving the Source 63
Returning to the Regular View 64
LESSON #7: EDITING A PAGE 64
LESSON #8: BOOKMARKS 64
Adding a Bookmark 65
Viewing Bookmarks 65
LESSON #9: THE INDEX 65
LESSON #10: SEARCHING 67
Page Searching 67
Index Documents 67
LESSON #11: SAVING A MULTIMEDIA FILE 67
LESSON #12: FILLING OUT FORMS 68
The Form of a Form 68
User Validation 70

LESSON #13: USENET NEWS POSTING 70
Selecting a Group 70
Browsing Articles 70
Reading an Article 71
Posting an Article 72
LESSON #14: SENDING E-MAIL 72
LESSON #15: THE LYNX OPTIONS MENU 73
Editor 75
Bookmark File 75
Personal Mail Address 75
Searching Type 75
VI Keys 75
Emacs Keys 76
Keypad As Arrows or Numbered Links 76
User Mode 76
Local Execution Scripts or Links 76
WHAT NOW? 77

4 SLIPKNOT 79

TYING THE KNOT: INSTALLING 82
HOW SLIPKNOT SLIPS 83
SETTING UP THE TERMINAL 83
Setting Up the Host 83
Setting Up Communications 87
Setting Up the Terminal 87
LESSON #1: CONNECTING 88
Logging In 88
Logging Out 88
LESSON #2: NAVIGATING 89
Following a Link 90
Snagging a Particular URL 91
Stopping a Load 91
Loading a Local File 91
Retrieve Again! 92
LESSON #3: TURNING OFF IMAGES 92
LESSON #4: RETRIEVING MULTIMEDIA FILES 92
Viewing 93
Saving 93
FTP Files 93
LESSON #5: YOU'RE HISTORY 93
Cleaning Up 94
Cleaning Up for Good 94
LESSON #6: THE PERMANENT SAVE 95
Local Keepers: Saving a Complete Web Page 95
Saving a Page 96
Remote Keepers: Bookmarks 96
Placing a Bookmark 96
LESSON #7: CHECKING OUT THE SOURCE 97
LESSON #8: CONFIGURING VIEWERS 97

Editing LView or WPLANY 97
Adding a New Viewer 98
LESSON #9: MAKING IT LOOK PRETTY 98
Setting Colors 99
Customizing Fonts 99
Background 101
LESSON #10: SWITCHING TO THE TERMINAL 102
LESSON #11: UPGRADING 102
WHAT NOW? 102

5 NCSA MOSAIC 103
INSTALLING MOSAIC 106
Installing WIN32s-OLE 106
Installing the Mosaic Browser 106
LESSON #1: NAVIGATING USING THE VIEW WINDOW 107
Following Hyperlinks 108
Moving to a Specific URL 109
The Right Mouse Button 109
Halting: The Secret Button 110
LESSON #2: IGNORING IMAGES 111
LESSON #3: MULTIMEDIA FILES 112
LESSON #4: HISTORY LIST 112
LESSON #5: THE HOTLIST 112
Getting Hot: Quickly Creating a Hot Item 114
Customizing Your Mosaic Menu 114
The Quicklist 116
Sharing Hotlists 116
LESSON #6: KEEPING THE PAGE 116
Save It 117
Load It to Disk 117
Print It 117
LESSON #7: SEARCHING 118
LESSON #8: THE SOURCE OF IT ALL 118
LESSON #9: FORMS 118
LESSON #10: ANNOTATING 121
LESSON #11: GOOD NEWS AND BAD NEWS 121
LESSON #12: FTP 122
LESSON #13: E-MAIL 123
Sending Mail 123
Mail to Developers 123
LESSON #14: PRESENTATION MODE 123
LESSON #15: VIEWERS AND OTHER HELPERS 124
LESSON #16: OTHER MOSAIC OPTIONS 125
The Fonts You Want 125
Colors 125
Home Sweet Home 126
WHAT NOW? 126

6 THE NETSCAPE NAVIGATOR 127

INSTALLING NETSCAPE 131
LESSON #1: SCAPING THE NET 131
Following Hyperlinks 133
Moving to a Specific URL 133
Loading Indicators 134
Stop It! 134
The Web Page Appears 134
Safe and Secure 134
LESSON #2: GRAPHICLESS: NO AUTO LOADING 135
Loading Later 136
Hither and Dither 136
LESSON #3: RETRIEVING MULTIMEDIA FILES 137
LESSON #4: ANCIENT HISTORY 137
LESSON #5: BOOKMARKS 138
Dealing with Bookmarks 139
Organi zingBookmarks 141
The Bookmark Files 143
LESSON #6: KEEPING A WEB PAGE 144
Saving to Disk 144
Loading 144
Printing 145
Mail Document 145
LESSON #7: SEARCHING A PAGE 146
LESSON #8: GETTING TO THE SOURCE 146
LESSON #9: FILLING IN FORMS 147
Filling 'er Out 149
User Validation 149
LESSON #10: THE NEWS 149
Setting Up a News Ser ver 150
Getting the News 150
Su bscribing to Newsgroups 151
Browsing Articles 151
Reading an Article 152
Posting an Article 154
LESSON #11: E-MAILING 154
LESSON #12: GRABBING FTP FILES 156
LESSON #13: CONFIGURING VIEWERS 157
LESSON #14: CUSTOMIZING YOUR NETSCAPE EXPERIENCE 159
You've Got the Look 159
There's No Place Like Home 160
WHAT NOW? 160

7 A BRIEF LOOK AT OTHER BROWSERS 161

SPRY AIR MOSAIC 164
Fea tures 164
Getting It 165
WINWEB 166
Features 166
Getting It 167

CELLO 168
Features 169
Getting It 169
SPYGLASS ENHANCED MOSAIC 169
Features 169
Getting It 170
NETCRUISER'S WEB BROWSER 170
Features 170
Getting It 171
WHAT NOW? 171

PART II CREATING WEB PAGES 173

8 WHAT CAN I DO? 173
NAVIGATE: PLACES TO START 176
Best of the Web 176
Global Network Navigator 176
Einet Galaxy 176
Yahoo 178
INFORM: TERRIFIC TEXT 179
Manuals: NCSA Mosaic 179
Helpful Information: OncoLink 180
A New Type of Text 180
HOME-MAKING: NO PLACE LIKE HOME PAGES 181
Corporate Pages 181
University Home Pages 183
Government Home Pages 185
Personal Home Pages 186
TEACH: EDUCATION 188
Geography Skills 188
ERIC 189
Britannica Online 189
ENTERTAIN: GROOVY GRAPHICS 190
TV: Interactive TV Index 190
Movies: Buena Vista 190
Music: Underground Music Archive 190
Art: ArtSource 192
Sports: Internet Baseball Information Center 192
PUBLISH: PAPERS, BOOKS, MAGAZINES, AND MORE 192
Academic Papers 193
Multimedia Magazines 194
'Zines 195
Books 196
SHOWCASE: PUTTING YOUR TALENTS ONLINE 196
SELL: TAKING CARE OF BUSINESS 198
Stock Quotes 198
Networking Over the Net 199
Job Searching 199
BUY: SHOPPING 199

The Internet Shopping Network 200
Direct Ordering 200
SPEAK FREE: POLITICS 202
PeaceNet 202
FedWorld 202
OBSERVE: THE NEWS 203
Headline News 203
Weather 204
INTERACT: FILL-IN FORMS 204
Personal Ads 204
Databases 204
Security 205
FIND: MAPS 205
Interactive Weather Map 205
Xerox Map Server 206
Map of Mars 206
TOUR: MUSEUMS 208
Le WebLouvre 208
Exploratorium 209
Museum of Paleontology 209
VISIT: VIRTUAL PLACES 210
San Diego 210
Paris 211
EXPO Ticket Office 212
Downtown Anywhere 213
Internet Town Hall 214
BROWSE: LIBRARIES 214
SURPRISE: OTHER NEAT GIMMICKS 216
LabCam 216
The Cyrano Server 217
Lite-Brite 217
The Virtual Frog 218
Say 218
Bianca's Smut Shack 219
A Puzzler 220
WHAT NOW? 220

9 THE GAME PLAN 223

ENTER THE WEBMASTER 225
LESSON #1: WHO'S YOUR AUDIENCE? 226
Does the Audience Exist? 226
What's Already Out There? 227
Focusing in 227
LESSON #2: WHAT'S YOUR OBJECTIVE? 229
Sound and Video 230
Talking Back 230
Security 230
LESSON #3: THE HARD SELL 230
Astound! 231

Inform! 232
Entertain! 232
LESSON #4: OVERALL DESIGN 232
Point of Entry 233
The Home Page 233
Permanence 235
LESSON #5: PROBLEM PAGES 236
Art Bombardment 236
Graphic Gluttony 236
Multimedia Moderation 237
Textual Terseness 237
LESSON #6: GETTING THE RIGHT LOOK 237
Consistent Controls 238
The Look Should Fit the Structure 239
Cross-Browser Dressing 240
LESSON #7: LINKS TO OTHER PAGES 240
Stale Links 241
Senseless Links 241
LESSON #8: STUDYING THE STATS 242
WHAT NOW? 242

10 THE HYPERTEXT MARKUP LANGUAGE 243
WHERE TO START 246
Steal This Text 246
Convert This Text 248
HTML HISTORY 248
HTML FUNCTIONALITY 249
LIFE, LIBERTY, AND GOOD HTML 250
LESSON #1: HTML ELEMENTS 252
Specifying HTML 253
The Header 253
The Body 255
THE FUTURE OF HTML 256
WHAT NOW? 256

11 TEXT 259
DESIGN FUNDAMENTALS 262
Consistency 263
Signing Your Page 263
LESSON #1: SPECIAL CHARACTERS 263
Punctuation 263
Enhanced Characters 264
Other Characters 266
LESSON #2: SECTION HEADINGS 267
Style: Drafting Good Headings 269
Aligning Headings 270
LESSON #3: PARAGRAPHS 270
LESSON #4: LINE BREAKS 272
No Break 273

Word Break 273
LESSON #5: TEXT STYLE 273
Physical Styles 274
Logical Styles 275
LESSON #6: SECTION STYLE 276
Preformatted Text 276
Blockquotes 278
Addresses 278
Styles to Watch Out For 279
LESSON #7: FONT SIZE 281
Relative Font Size 282
The Base Font 282
LESSON #8: CENTERING 282
LESSON #9: COMMENTS 283
LESSON #10: LISTS 283
Ordered Lists 284
Unordered Lists 286
Customizing Lists 288
Definition Lists 288
Plain Lists 290
Menu Lists 291
Directory Lists 291
LESSON #11: HORIZONTAL LINES 292
In Thickness and in Health 292
Setting the Line's Width 292
Aligning Your Lines 293
Using a Solid Bar 293
LESSON #12: TABLES 294
Text Tables 294
Graphical Tables 295
HTML+ Tables 295
LESSON #13: HYPERTEXT 301
The Anchor Element 301
Links: HREF 303
Anchors: The Name Attribute 306
Title 309
Defining Relationships 309
Methods 311
LESSON #14: RELATIONSHIPS AND HYPERPATHS 311
Using Link 312
Nodes 313
Path 313
LESSON #15: MATHEMATICAL EQUATIONS 313
LESSON #16: DYNAMIC DOCUMENTS: PUSH OR PULL? 315
Server Push 315
Client Pull 315
LESSON #17: POLISHING 316
WHAT NOW? 317

12 GRAPHICS 319
GRAPHIC FORMATS 322
GIF 322
JPEG 323
GRAPHIC VIEWERS 323
LView 323
WinGIF 325
LESSON #1: INLINED IMAGES 326
The IMG Command 327
Distant Images 327
Hyperimages 327
Crossing the Border 328
LESSON #2: ALIGNMENT 328
Absolute Alignment 329
Floating Alignment 330
Mino in the Gutter 331
Clearing Space After Line Breaks 331
LESSON #3: TEXT ALTERNATIVES 332
LESSON #4: IMAGE SIZING 333
Auto-Scaling 334
Percentage Auto-Scaling 335
LESSON #5: TRANSPARENT GIFS 336
STEP 1: Isolate the Background 336
STEP 2: Selecting a Background Color 338
STEP 3: Making the Background Transparent 339
LESSON #6: CREATING SPACERS 342
LESSON #7: PERKS, GIZMOS, AND FRILLS 343
Bullets 343
Buttons / Icons 344
Bars 346
LESSON #8: MAKING GRAPHICS HUMBLE 346
Shrink Me 347
Thumbnails 348
The Flip Trick 350
LESSON #9: INTERLACED GIF IMAGES 351
LESSON #10: CLICKABLE IMAGEMAPS 352
STEP 1: Create the Image 353
STEP 2: Create the Map 354
MapEdit 1.1.2 355
STEP 3: Write the HTML 358
STEP 4: Set Up the Server 359
STEP 5: The Text-Based Index 360
LESSON #11: WHAT'S YOUR BACKGROUND? 361
Background Color 362
Foreground Color 362
Getting Backgrounds 363
LESSON #12: EMBEDDING 363
THE SILVER SCREEN 363
Lights, Camera... Action? 364

QuickTime 365
MPEGPLAY 365
GRABBING GRAPHICS 366
Playing Picasso 366
Capturing 367
Image Hunting 368
WHAT NOW? 372

13 SOUND 373
HEARING AIDS 376
Installing SPEAK.EXE 377
Configuring SPEAKER.DRV 377
JAMMIN' 378
WPLANY 378
WHAM 379
LESSON #1: HYPERSOUND 380
SOUNDING OFF 382
Audio Formats 382
Recording It Yourself 383
Searching for That Right Sound 384
WHAT NOW? 385

14 INTERACTIVITY 387
HOW A FORM WORKS 390
LESSON #1: A FORM IS BORN 392
ACTION 392
METHOD 392
ENCTYPE 393
LESSON #2: INPUT FIELDS 393
TYPE 394
NAME 395
DISABLED 396
MIN/MAX 396
LESSON #3: TEXT FIELDS 396
Text Field Variations 396
VALUE 398
SIZE 398
MAXLENGTH 398
LESSON #4: CHECKBOXES OR RADIO BUTTONS 398
VALUE 398
Checkboxes 399
Radio Buttons 400
Prechecked 400
LESSON #5: THE RESET AND SUBMIT BUTTONS 401
Customizing the Label 401
Using Icons 402
LESSON #6: TEXTAREAS 402
Size 403
WRAP 403

LESSON #7: SELECTION LISTS 403
SIZE 404
Multiple Selections 405
Options Attributes 405
LESSON #8: PUTTING IT ALL TOGETHER 406
A Basic Form 407
A Sample Message Form 407
LESSON #9: INCLUDING HIDDEN TEXT 408
LESSON #10: SEARCH INDEXES 410
ISINDEX 411
HREF 411
PROMPT 411
Book-Like Index Documents 411
LESSON #11: TESTING IT OUT 412
LESSON #12: WHAT THE DATA LOOKS LIKE 412
Search Indexes 413
The GET Method 413
POST 414
THE FUTURE OF FORMS 414
WHAT NOW? 415

15 CGI SCRIPTS 417
LESSON #1: WHAT IS CGI? 420
What Can I Do with a CGI Script? 422
LESSON #2: THE STANDARD CGI SCRIPTS 423
LESSON #3: HOW TO USE PRE-EXISTING SCRIPTS 426
Constructing a CGI URL 426
Specifying a Script Within a Form 427
Specifying a Script from the "Open URL" Interface 427
LESSON #4: HOW TO USE A SCRIPT TO ACCESS OTHER APPLICATIONS 428
LESSON #5: WRITING YOUR OWN SCRIPTS 430
Writing a Simple Script 431
Using Environment Variables in Scripts 435
Location and Status Headers 437
Security with CGI Scripts 439
LESSON #6: HANDLING FORM DATA 441
What Does Form Data Look Like? 441
LESSON #7: SAMPLE SCRIPTS FOR UNIX, WINDOWS, AND MACINTOSH SERVERS 451
Unix 451
Neat Tricks for HTML 3 Browsers 457
DOS/Windows Scripts 460
Macintosh Scripts 462
WHAT NOW? 465

16 OTHER WEB RESOURCES 467
LESSON #1: GOPHER 470
LESSON #2: NET NEWS 472
LESSON #3: E-MAIL 474
LESSON #4: WIDE AREA INFORMATION SERVICE 477

LESSON #5: FILE TRANSFER PROTOCOL 478
What Happens to Downloaded Files? 479
How Can I Send a File Using FTP? 481
LESSON #6: TELNET 481
What Does Telnet Allow Me to Do? 481
What Happens When I Initiate a Telnet Session? 481
LESSON #7: ACCESSING LOCAL FILES 482
LESSON #8: URL SCHEMES 483
WHAT NOW?: 483

17 CONVERTING, TRANSLATING, OR CHEATING 485

WORDPERFECT TO HTML 489
Installing It 489
Writing Your Document 490
Converting 491
MICROSOFT WORD TO HTML 491
ANT_HTML 491
CU_HTML 497
The Internet Assistant 501
Other Tools 501
BOOKMARKS TO HTML 501
Lynx 502
Mosiac 502
Netscape 502
Cello 502
WHAT NOW? 502

18 HTML ASSISTANT 505

INSTALLING THE ASSISTANT 508
LESSON #1: FROM TEXT TO HTML 508
LESSON #2: CREATING FROM SCRATCH 510
LESSON #3: THE TOOLBAR 510
Selection Tools 511
Insertion Tools 513
LESSON #4: INSERTING HYPERLINKS 513
External Links 514
Internal Links 515
Recycling URLs 516
LESSON #5: IMAGES 516
LESSON #6: USER TOOLS 516
Creating a User Tool 517
Entering Enter 518
LESSON #7: FINDING OR REPLACING 518
Finding 518
Replacing 518
LESSON #8: TOOLING WITH TOOLS 518
Undo 519
Auto Repeat 519
Putting an HTML Document on Hold 519
LESSON #9: TESTING 'ER OUT 519

Specifying a Browser 520
The Final Test 520
LESSON #10: PRINTING 520
LESSON #11: URL FILES 520
Grabbing URLs 521
Editing URLs 521
Combining URL Files 522
Saving It 522
LESSON #12: CONVERTING A BOOKMARK, HOTLIST, OR URL FILE TO HTML 522
WHAT NOW? 523

19 HoTMetaL 525

INSTALLING IT 528
LESSON #1: STARTING WITH A TEXT FILE 529
LESSON #2: STARTING FROM SCRATCH 529
Templates 530
Saving 531
LESSON #3: MARKING IT UP 531
Inserting 532
Surrounding 535
Changing 535
Deleting 536
The Smart List 536
Pinning 536
LESSON #4: SPECIAL CHARACTERS 537
LESSON #5: WORKING WITH HYPERLINKS 538
Creating External Links 538
Creating Internal Links 539
LESSON #6: WORKING WITH GRAPHICS 540
Image Attributes 540
Viewing Images 541
LESSON #7: WORKING WITH FORMS 541
Input Field 541
TEXTAREA 543
Selection Lists 544
LESSON #8: FINDING SOMETHING 545
The Search Term 545
Search Options 546
Searching 547
LESSON #9: BREAKING THE RULES 547
LESSON #10: STYLES 547
Character 548
Separation 549
Load Styles 549
LESSON #11: TESTING IT OUT 550
Hiding Tags 550
Publishing 551
Previewing 551
LESSON #12: THE HTML BACKBONE 551
The Context Window 552

The Structure Window 552
WHAT NOW? 553

PART III WEAVING A WEB OF YOUR OWN 555

20 WHERE TO PLACE YOUR HTML DOCUMENTS 555
LESSON #1: TROUBLESHOOTING YOUR HTML DOCUMENTS 558
LESSON #2: HTML DOCUMENT ANALYSIS SERVICES 560
LESSON #3: FREE WEB POSTING SPOTS 563
LESSON #4: WEB SPACE FOR SALE 565
LESSON #5: POSTING 570
LESSON #6: UPLOADING DOCUMENTS 572
Uploading via FTP 573
Transferring Files with a Modem 574
LESSON #7: INSTALLING PAGES ON AN EXISTING WEB SERVER 574
WHAT NOW? 576

21 STARTING YOUR OWN WEB SITE 579
LESSON #1: YOUR WEB SERVER 582
LESSON #2: WEB SERVER SOFTWARE 582
LESSON #3: SERVER SOFTWARE SOURCES 585
LESSON #4: TIPS AND CAUTIONS 586
Type of Machine 587
Memory 587
LESSON #5: SECURITY! 588
Transmission Security and Data Encryption 589
Access Control, Passwords, and Other Filters 591
LESSON #6: HARDWARE 598
LESSON #7: INSTALLING A WEB SERVER 599
Unix Server Installation 599
Windows Server Installation 613
Macintosh Server Installation 623
LESSON #8: HYPERMEDIA GATEWAYS 628
WHAT NOW? 628

A SLIP INTERNET PROVIDERS 631

B OTHER SOFTWARE SOURCES 637

C WEB HELP RESOURCES 647

D ABOUT THE CD 653

INDEX 657

ACKNOWLEDGMENTS

First the technical stuff... Thanks to Tim Berners-Lee and CERN, the fore-folks of the Web. Thanks to NCSA for tiling cyberspace with their Mosaic. Thanks much to the brilliant authors of the software on the CD; the Web'd be empty without ya. Thanks to New York University, the Media Research Lab, and the Center for Advanced Technology for their excellent resources. Thanks also to Ken Perlin, Gus Estrella, and Robert Denny.

Now the personal stuff... Thanks to the friends whose nights about town I've forsaken while writing this. Thanks to Mom, Pop, and Jay (who loves to see his name in print). Thanks to Bob Dylan for the soundtrack. Thanks to Dan Scherf, who, as usual, saw this project through nearly hitchless and had a bit of fun besides (I hope!). Thanks to Jill Pisoni and Mitch Waite, who helped develop this book into the powerhouse it is. Heck—let's be fair—thanks to all the Waite Group Press for the tremendous job they did.

INTRODUCTION

Ah, the beauty, the sheer beauty of this string of weird words:

```
http://found.cs.nyu.edu/graphics.b./student/dsf4185/html/
```

The address of my World Wide Web home page.

It took several months, from start to finish, to get the page looking the way I wanted. It started one way and ended completely different. But in the end, I loved the result. It had writing samples, artwork, descriptions of my current projects, lists of my specialties, and an easy way of letting folks visit my favorite sites around the Web. Getting my Web pages working was equal to the thrill of publishing my first book. Here was all the information I could hope to share—free, open, and offered to an audience of nearly 30 million people.

I wasn't publicizing my home page, so only a dozen or so users a day would actually stumble there. But still, that's a dozen people who know my name, who know what I'm about, who know where and how and why to reach me. And, more importantly, a dozen more people to whom I've given the gift of information or entertainment.

WHO THIS BOOK IS FOR

This is actually three books in one, as evidenced by the three sections:

HTML A book for people who want to learn about what the Web is, how to access it, and how to navigate around using a number of popular tools. Maybe you've never heard of the Web before, or maybe you're a seasoned ol' net-dog hoping to learn some new tricks. Either way, the first section of this book, Connecting to the Web, has what you

need. Popular software such as Winsock, The Internet Adapter, Lynx, SlipKnot, NCSA Mosaic, and Netscape are covered in full.

HTML A toolbox and guide for artists, advertisers, marketers, programmers, or professional "webspinners" who want to develop really cool stuff to put on the Web. Most likely, you think creating a Web page looks way too complicated. This book shows you that that's the furthest thing from the truth. Perhaps you've already fooled a bit with Web production and creation, but have no idea how to achieve that professional look you've seen elsewhere. Or perhaps you just want a handy reference bible in your tool belt. This section has it all, from the simplest detail to the latest hypertext specs.

HTML A tutorial on starting your own Web site, whether you're a home hobbyist or an international conglomerate. Maybe you want to start a commercial Web, renting your computer space to other Web developers. Or maybe you just want to learn about how the guts of the World Wide Web really work.

WHY THIS BOOK IS DIFFERENT

Although the Web has only recently become popular, there are many books clamoring for your attention. Some books discuss the theory and history of the Web. There are books showing you exactly how to use the NCSA Mosaic Web browser. Many act as travel-guides, indexing popular places on the Web. A few may talk about the language used to create Web pages: The Hypertext Markup Language.

This book does all of the above and more. The *Web Publisher's Construction Kit* takes you way beyond the basics. *Every* popular World Wide Web browser is covered fully, including Lynx, SlipKnot, Mosaic, and Netscape. You will learn the ins and outs of text, hypertext, artwork, fonts, images, movies, sounds, interactive forms, custom programs. This book doesn't stop at technical commands and how-tos, but it looks beneath the hood at creative issues. When you put down this book, your head should be overflowing with tons of cool ideas, along with the knowledge to implement them quickly and easily.

Further, this book delves into the mysteries of creating a Web site of your very own. This involves knowing more than what software to install and

which commands to type. It also requires a lot of keen skill, savvy, and creativity.

Most importantly, all the Web browsers, Web servers, development tools, hypertext editors, scripts, clip art, sounds, movies, or other media you need to accomplish these goals are bundled with the book on one jam-packed CD-ROM.

All in all, this book doesn't just teach you to use the Web, put together a Web page, and create a Web site of your own; it teaches you how to do it professionally, slickly, and in complete detail. In the end, your efforts will clearly stand out in a Web which is already overpopulated with lots of boring or lackluster corners.

WHAT THIS BOOK ASSUMES YOU KNOW

This book assumes you're vaguely familiar with the online world, the Internet, and Microsoft Windows. If you've never messed with a modem, or you think Unix are the people who guard harems (sorry, a very old joke), then you should check out a good, basic Internet book, such as The Waite Group's *Internet How-To*.

Other than that, chapters in this book each have a number of lessons. Easy concepts are covered first, more difficult and obscure things come later. You can read enough to become proficient, or you can skip the easy parts and jump right into expert design, implementation, and production issues.

However you use this book, I look forward to seeing you around the Web...

THE CHAMELEON SAMPLER

The NetManage Internet Chameleon is one of the most versatile and easy-to-use set of Internet tools in the world. Chameleon helps you sign up with an Internet provider, connect cleanly to the Internet, and access a variety of resources—including a pretty cool Web browser. The Chameleon package includes

- HTML *Custom,* for connecting to the Internet
- HTML *WebSurfer,* a full-featured World Wide Web browser
- HTML *Gopher,* which lets you access any gopher menu worldwide
- HTML *NEWTNews,* a Usenet newsreader
- HTML *Mail,* a convenient way to send and receive e-mail
- HTML *Archie,* which lets you search for a file over the Internet
- HTML *Telnet,* for connecting to a remote computer
- HTML *FTP,* for transferring files over the Internet
- HTML *FTP Server,* which lets you allow others to download or upload files to your PC
- HTML *Mail Utilities,* programs that help you compact or organize your mailbox files to save space
- HTML *Ping,* to test if you're connected to a remote computer
- HTML *Finger,* to check if a friend is connected to the Internet
- HTML *Whois,* to get information about people registered in the NIC (Network Information Center) database

You can sample the Chameleon tools for 30 days at no charge. If you like what you see, you can register everything for 50 bucks.

INSTALLING THE CHAMELEON

NOTE: In the installation directions here, we assume that your hard disk is the C: drive and your CD-ROM is the D: drive. If this doesn't match your computer, substitute C: or D: with the correct drive designation.

To copy the sampler software onto your hard disk, run the Setup program. While under Windows, select File, Run in the Program Manager. In the Run dialog box, type

```
d:\windows\browsers\ntmanage\disk_1\setup.exe
```

and then press the OK button.

The Setup program will ask you where to install the NetManage program. The default suggested is fine for most people. If you want it installed elsewhere, type in the drive and directory of your choosing and select Continue.

After a few moments, the Setup program will ask you to type in the path of the second batch of files. Select the *1* in *DISK_1* and change it to 2, and select Continue.

After another few moments, the Setup program will ask you to type in the path of the third batch of files. Select the *2* in *DISK_2* and change it to 3, and select Continue.

Click OK when Setup tells you that installation is complete. You are now ready to setup your Internet account!

SIGNING UP FOR AN INTERNET PROVIDER ACCOUNT

If you don't already have one, the Chameleon package makes it easy to sign up with one of several popular Internet providers. Read Chapter 1 for more information about what services are offered by Internet providers.

If you'd like to sign up using the Chameleon software, run the Automatic Internet-Click to Start icon.

To learn about a particular Internet provider, click one of the tabs (other than NetManage) in the Select Internet Provider window. Most providers give you several hours (or even a month) of free trial time. To read about the

Figure I-1 The easiest way to sign up for an Internet provider

locations an Internet provider can cover, the monthly price, and other important information, click the More Info button at the bottom of the screen. If you have specific questions, contact the provider directly.

When you're ready to begin the sign-up procedure, click the Signup button. You'll see a registration screen similar to the one in Figure I-1. Fill in your name (as it appears on your credit card), address, phone number, and credit card information.

NOTE: You will not actually be charged any provider fees until you officially register with the service. You can cancel the registration transaction at any time during the sign-on process. If you do decide to register, your credit card number will be sent over a secure phone line.

As you work through the sign-up process, there may be other tabs asking for additional information. If so, click these tabs and fill in the forms.

Select the Phone List button at the bottom of the screen. The Phone List dialog appears, listing possible phone numbers you can use to register. If one of the numbers is in your area code, select it. Otherwise, select the toll-free 800 number.

NOTE: If necessary, you can edit the registration phone number. Some systems, for example, require you to dial a 9 to reach an outside line. Just type in this 9.

When you've typed in all your vital stats, return to the first registration tab. Click Send to dial the toll-free number and begin the registration process. The icons to the right will light up as each stage of the dialing process is completed. The set of traffic lights tell you if each stage—initializing the modem, dialing, connecting, and communicating—has worked.

NOTE: You may need to click the Advanced button to specify special modem ports or commands.

Follow the instructions that appear as the registration proceeds. You will be given the option to select from various service and pricing plans. Your account information (username, e-mail address, password, dial-up number, and IP address) will automatically be configured into the Chameleon package. An interface will be created for the Custom program, which quickly and flawlessly connects you to the Internet.

That's it! You can now reboot your system to kick-start everything.

REGISTERING THE CHAMELEON SOFTWARE

If you already have an Internet account, you can set up the Internet Chameleon software (shown in Figure I-2) and start using it within minutes. Run the Automatic Internet-Click to Start program.

Make sure the NetManage tab is selected, and then click the Signup button. You can now activate the software for a free 30-day demonstration period. After this period, the Chameleon software will no longer work. If you decide to register the Chameleon package (for $50), your credit card will be charged and your software will be activated permanently.

Fill in all the information on both forms, as shown in Figure I-1, including your credit card number (it won't be charged unless you complete the registration). You may need to contact your Internet provider for the Internet information on the second form.

Select the Phone button, and choose a local or toll-free phone number. Then click the Send button to dial in to NetManage and get your software activated.

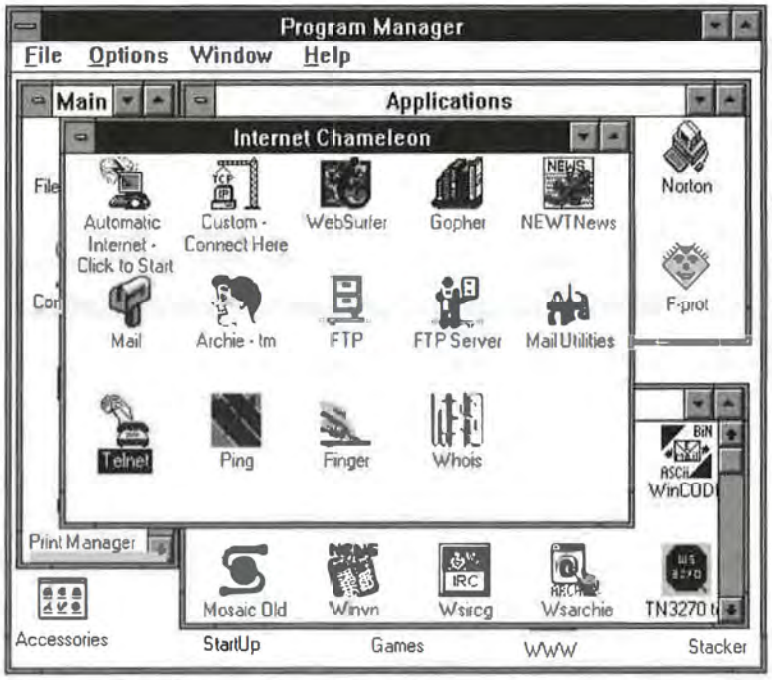


Figure I-2 The full Chameleon package in the Internet Chameleon program group

Once you connect, you are given the following choices:

HTML Activate your software for a free 30-day demonstration.

HTML Purchase your software to activate it permanently.

HTML Configure your connection (if your Chameleon software has already been activated).

CONNECTING TO THE INTERNET

Now that you have selected a provider and registered your software, you can actually get hooked in to the Internet. To do this, you need to run the Custom program (Figure I-3) from Windows File Manager.

If you used the Chameleon package to sign up with your Internet provider, an automatic configuration file should have already been written

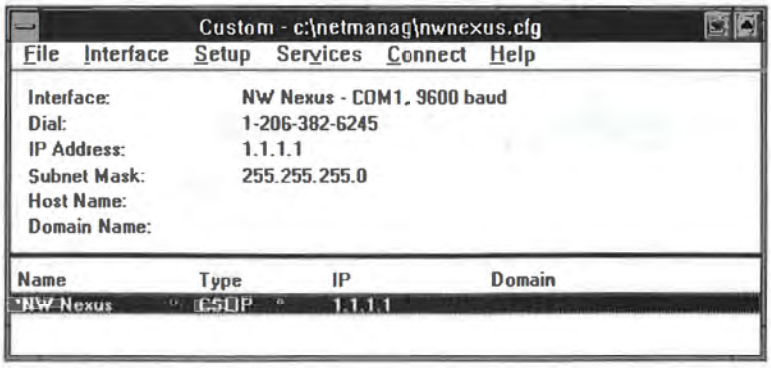


Figure I-3 Your customized on-ramp onto the Information Superhighway

for you. Otherwise, Chameleon comes with the configurations for most popular Internet providers. Select File, Open and look for the configuration file for your provider. If your provider is not listed, you'll need to contact them and ask what the proper settings are. They may even be able to send you a prewritten Chameleon configuration file.

If you do need to enter the connection settings yourself, use the appropriate values you have obtained from your Internet provider. You can verify or edit the following information under the Setup menu:

- HTML IP Address
- HTML Subnet Mask
- HTML Host Name
- HTML Domain Name
- HTML Port
- HTML Modem
- HTML Dial
- HTML Login
- HTML Interface Name

- HTML BOOTP

You may also need to fill in the following under the Services menu:

- HTML Default Gateway
- HTML Domain Servers

Read Chapter 1 for more information about these terms.

Logging In

Once your configuration settings are in place, simply click the Connect menu to dial up your Internet provider and get connected. If all goes well, you should hear a small beep, and a program known as Newt will run. This program lets Windows communicate with the Internet. You can then minimize the Custom program and run the Internet application of your choice.

Logging Out

When you're done using the Internet, call up the Custom program and click the Disconnect menu.

WEB BROWSING WITH WEBSURFER

WebSurfer is a full-featured World Wide Web browser similar to Mosaic. You can read all about browsers in Chapter 2 and about Mosaic in Chapter 5. To start exploring the Web, first use the Chameleon Custom program to connect to the Internet. Then run the WebSurfer program.

Like Mosaic, WebSurfer has a toolbar (see the top of Figure I-4) that acts as a shortcut for most commands. The toolbar contains

- HTML Show Connection Status: Shows you which links are currently being loaded.
- HTML Go to URL: Opens a specific Web URL (defined in Chapter 1).
- HTML Get URL: Reloads the current document.
- HTML Hotlist: Shows the list of your favorite Web pages for you to choose from. To go to a page, just double-click on it. You can also delete pages from the list by selecting the page and clicking Remove.

HTML Make Hot: Adds the current Web page to your hotlist.

HTML Back: Revisits the Web page you just came from.

HTML Forward: Goes to the next Web page in the series, if applicable.

HTML Home: Returns to the Web page you started from.

HTML Cancel All: Stops the loading of the current Web page.

Loading a Web Page From the Internet

Like Mosaic, WebSurfer combines text and graphics on the same page. Any text in blue or graphics with a blue border are hypertext links to other Web pages, multimedia files, or Internet areas. To load a link, just click on it.

You can also load up a document directly. Just select Retrieve, Go To URL and type in the document's exact URL. Alternatively, you can type a document's URL in the Dialog bar's URL box and press [ENTER] to load it.

If the document is a Web page, it will be displayed. If the document is a graphic, sound, or movie, the WebSurfer program will attempt to call up a viewer program to display/play it. If the document is any other type of multimedia file, WebSurfer allows you to save the document directly to your hard disk.

To find out more about the current Web document, select Retrieve, Properties.

Loading a Web Page From Your Hard Disk

If you have any Web pages on your hard disk (perhaps ones that you've created yourself), you can easily use WebSurfer to view them. Select Retrieve, Open Local File. Choose the file you want to view and click OK.

You can even edit the current Web document—a very handy capability for Web developers. Select Retrieve, Edit HTML. Then access the Retrieve, Refresh From Disk menu item to reload the page in a flash and see what your edits look like.

OTHER INTERNET TOOLS

The Chameleon package contains software for every Internet resource you could possibly want. To use FTP, e-mail, telnet, gopher, or any other Internet program, first connect to the Internet using the Custom application. Then

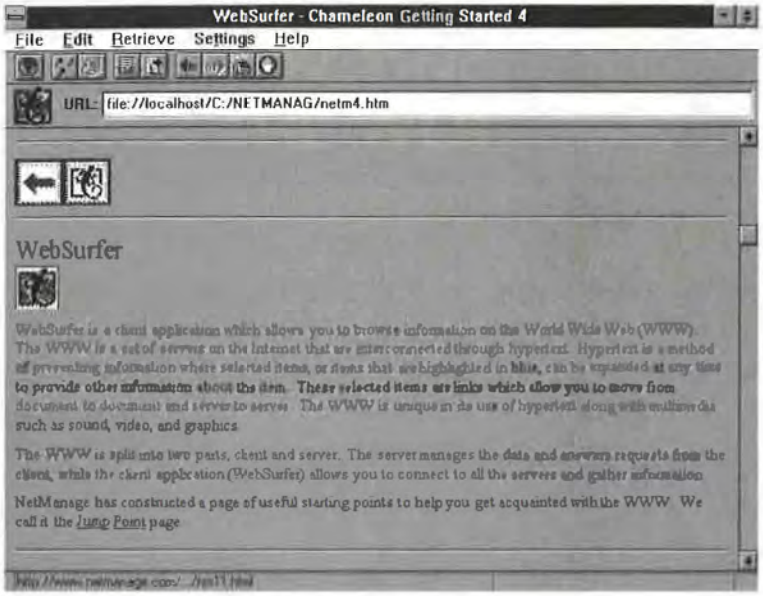
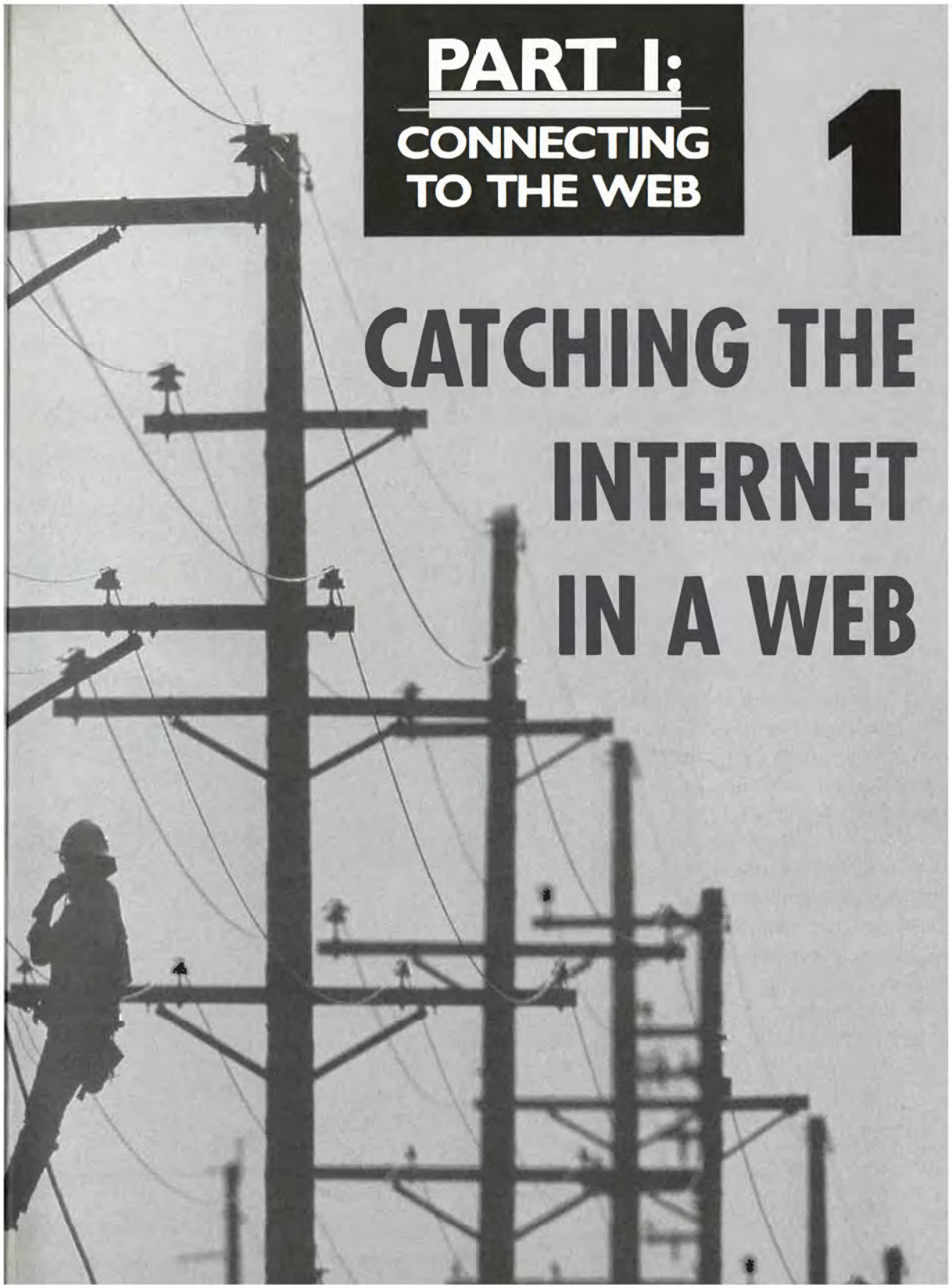


Figure I-4 The WebSurfer browser in all its glory

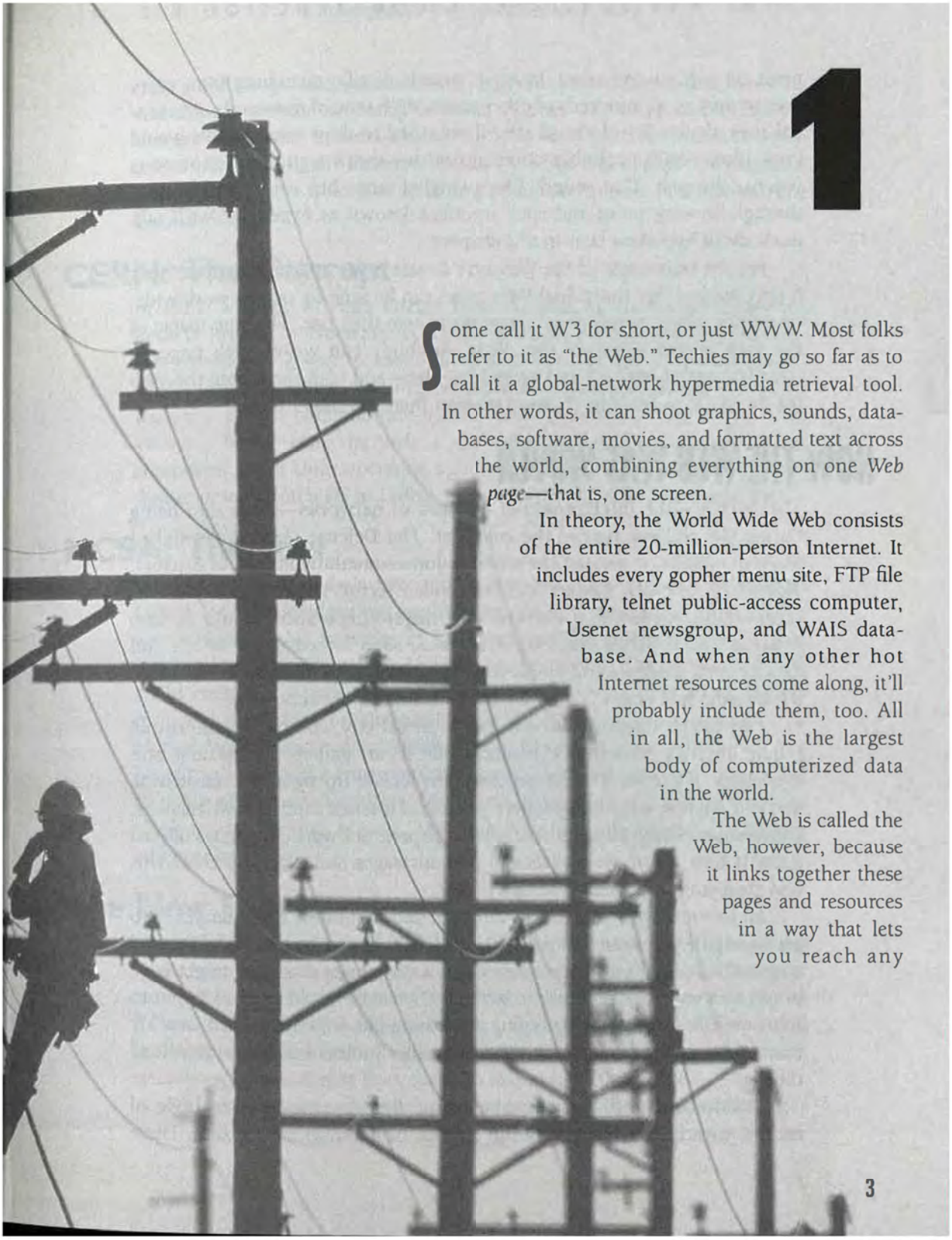
you can communicate with friends across the world using Mail, read or post messages to thousands of newsgroups using NEWTNews, browse menus of data using gopher, download tons of cool software using FTP, and much more.



PART I: CONNECTING TO THE WEB

1

CATCHING THE INTERNET IN A WEB



1

Some call it W3 for short, or just WWW. Most folks refer to it as "the Web." Techies may go so far as to call it a global-network hypermedia retrieval tool. In other words, it can shoot graphics, sounds, databases, software, movies, and formatted text across the world, combining everything on one *Web page*—that is, one screen.

In theory, the World Wide Web consists of the entire 20-million-person Internet. It includes every gopher menu site, FTP file library, telnet public-access computer, Usenet newsgroup, and WAIS database. And when any other hot Internet resources come along, it'll probably include them, too. All in all, the Web is the largest body of computerized data in the world.

The Web is called the Web, however, because it links together these pages and resources in a way that lets you reach any

point on it from any other. Imagine strands of silk, stretching from every spot to dozens or hundreds of other spots. With tens of thousands of potential sites, that's a lot of virtual silk. If you tried to draw what all this would look like, you'd probably come up with—that's right—an enormous psychedelic web. That sounds like a tangled mess, but you can navigate it through an easy point-and-click interface known as *hypertext*. We'll talk more about hypertext later in this chapter.

But the true magic of the Web isn't its size or its multimedia snazziness. It isn't the fact that individual Web pages can be seen by people worldwide, no matter what computer or operating system they use. The true magic of the Web is that anybody can do it. Anybody can weave Web pages—complete with hypertext and everything else—and spin them onto the Web for the world to see. And doing it is easier than you might think.

HOW THE WEB WAS WOVEN

The *Internet*—the interconnected network of networks—came into being during the chilliest part of the cold war. The Defense Department linked together computers around the world to form something known as Arpanet. Slowly this network melted into the civilian sector. Looking to expand its communication power, a university or major corporation would tie into Arpanet, and students and employees were soon linked to the minds and data of other students and employees. The Net spread like a virus. By 1993, the Internet had nearly 15 million users in almost 70 countries.

The Internet rapidly turned into a global city, complete with virtual public libraries, storefronts, business offices, art galleries, museums, and even bars. There were some private sites, locked up tight, but most were open to anyone who knew where to look. Libraries offered card catalogs. Universities offered all sorts of academic papers. Software companies offered support files. There were places to post messages and places to chat. This was great stuff. It added up to a *lot* of text.

All this text was hard to sort through, and, frankly, it had a tendency to get boring. If you wanted a particular article, first you had to find what and where it was, then you had to figure out a way to get at it. You might have to use a newsreader program to grab it. Or maybe you'd need to log in to someone else's computer by typing something like *telnet somewhere.com*. Or maybe you would have to use the file transfer protocol (FTP) to download the file.

Utilities like gopher came into being. By selecting from a jungle of menus, you could retrieve files and articles, or log in to remote sites. There

were a few images and sounds online, but you needed specialized hardware and software to take advantage of them. Gopher was a step better than the barrage of Internet commands, but still far from perfect. Different documents were hard to sift through. It didn't feel like a newspaper or magazine or even a TV show, it felt like a blizzard of text and photos and files swirling around you.

CERN: The Concept

In 1989, a bunch of High Energy Physics dudes at the Conseil Europeen pour la Recherche Nucleaire (CERN)—led by an ambitious Tim Berners-Lee—developed a way to share all sorts of information over the Internet. CERN was an international organization, and members across the world wanted a way to communicate quickly and consistently. The World Wide Web was born. Using the Web, a team of scientists in Geneva using Sun computers and a Unix operating system could read about particle propulsion tests stored on a PC in Dallas, Texas. Everything could be hassle-free.

NCSA: The Tool

But it wasn't quite hassle-free until a few years later, when the National Center for Supercomputing Applications (NCSA) developed an attractive way to look at this universe of information. One lovely summer day in 1993, NCSA officially released *Mosaic* to the public. The Mosaic software could consistently and elegantly display Web graphics, text styles, hypertext, and other files on the same page like a sort of, yup, mosaic. The Mosaic Web browser was released as noncommercial software, for the benefit and use of the entire Internet community. Versions were created for all major platforms—Windows, Macintosh, Unix, Sun, and Silicon Graphics workstations. Before Mosaic, the Web was like a bunch of TV signals without the TV.

The New Fabric

Much like the Internet itself, the Web caught fire without really trying. New Web users put their own pages of information over the Web. This new variety of information attracted more Web readers, who in turn put more information on the Web, which in turn . . . you get the idea. Before 1993, the Web had about 100 official sites. After Mosaic came out, Web traffic steadily increased. A year later, the Web had nearly 10,000 sites.

As fast PCs and modems became less expensive, the everyday computer user could transfer large files such as graphics, movies, and sound. Soon

these media found their way onto the Web. The old fascinating but frustrating Internet suddenly became streamlined, organized, and really, really neat to look at.

And this isn't ancient history. The most exciting aspect of all this is that it *just* happened. The Web is in its infancy right now, and it's growing fast. It seems likely that it will serve as the backbone for tomorrow's video teleconferencing, interactive television, and shared virtual reality. You can be a cyberspace pioneer by weaving your own little Web corner today.

THE WEB'S EVOLUTIONARY EDGE

The World Wide Web isn't the first tool for fishing data out of the Internet, but it's the best so far—and it will keep on evolving. Here are the features that make the Web the global superhighway hypermedia information retrieval system of choice:

- The Web is simple. Once you're online with a program like Mosaic, all you need to access a world of data is a pair of eyes, a mouse, and a healthy curiosity.

- Documents look great. Instead of plowing through endless text files, you'll find a mix of fonts, styles, and layouts. You'll even get graphics and text together on the same screen, presented seamlessly.

- The Web keeps growing It can deal with gopher, FTP, telnet, and other Internet tools now, and built-in features will let it adapt as new standards, protocols, data types, and languages develop.

- The Web can be accessed anytime, anywhere, any which way. You can surf the Web from most any type of computer or operating system, and from any country and any part of the Internet.

- The Web lets you go anywhere even when you don't know where you're going! With hypermedia links, each Web page points to lots of others; you can skip around the Internet randomly, pursuing any topic that interests you from site to site.

- The Web has no owner. With the right equipment and software, anybody can read the Web or distribute data on it—even you.

All this sounds great, but how is it possible? Read on . . .

WORLD WIDE WEB LAYOUT

Most any type of network consists of computers known as *clients* and *servers*. The server is a powerful computer that does all the grunt work, holding most of the files, memory, data, and resources. Individual PCs usually act as clients. Though weaker and dumber, clients exploit the mighty server.

The Web can be thought of as a vast network of data. As such, it has its own servers and clients, as illustrated in Figure 1–1. The servers hold all the Web data and dish it out, much like publishers or TV stations. The clients can access the Web data itself. Client programs like Mosaic are called *Web browsers,* since you use them to browse through the Web.

To access the Web, you need to install a browser on your computer, turning it into a Web client. Your browser then uses the Internet to connect to another computer running the Web server software. The server serves up whichever Web page your client asks for. Your client then slaps the Web document onto your screen. The Web is nothing more than a tangle of these client-server interactions, all happening thousands of times a second.

Figure 1–1 Web servers distribute Web pages; Web clients bring them to users' computer screens

The Server

A Web server can be much smaller and simpler than you may expect. Most major Internet providers have their own Web servers. If you have a permanent Internet connection, you can even install a server on your own machine.

You can learn how to install and configure a Web server in Chapter 21, Starting Your Own Web Site. Chapter 20, Where To Place Your HTML Documents, discusses existing Web servers who can publish your Web pages for you.

The Client Browser

Since the advent of Mosaic, dozens of browsers have hit the market. Some are commercial, others are perfectly free. Each browser has its own perks, bells, and whistles. The next chapter discusses browsers in general. Chapter 5, NCSA Mosaic, is a hands-on tutorial of the Mosaic software. Other chapters discuss the popular Netscape browser, the Lynx text-only browser, and the revolutionary SlipKnot browser.

NAVIGATING WEB PAGES

A Web page can be anything you imagine it to be: a hypertext document, graphics, sound, animation, tabular data, software files, a discussion group, a search index, a specialized program, and more. Most every Web site has a standard, default page known as the *home page*. This page acts as a sort of front door. Anybody who accesses a site usually starts off at that site's home page.

Many people create personal home pages for themselves, including things like their photograph, current projects, addresses, hobbies, even complete resumes. These *author pages* stick a human face onto the Web.

Most browsers use uniform fonts and styles. This means every Web page will have a similar layout and look. Web pages can contain everything from academic articles to shopping catalogs to art galleries to travel guides. Chapter 8, What Can I Do?, is a photo-essay tour of some of the world's most popular Web pages. Some pages will haunt you, others will teach you, most will entertain you. In general, though, there are only two ways you can interact with a Web page:

HTML Click on a *hypermedia link* to access another page, program, or Internet service

HTML Search for a keyword within a *search index* or fill in input fields in a form

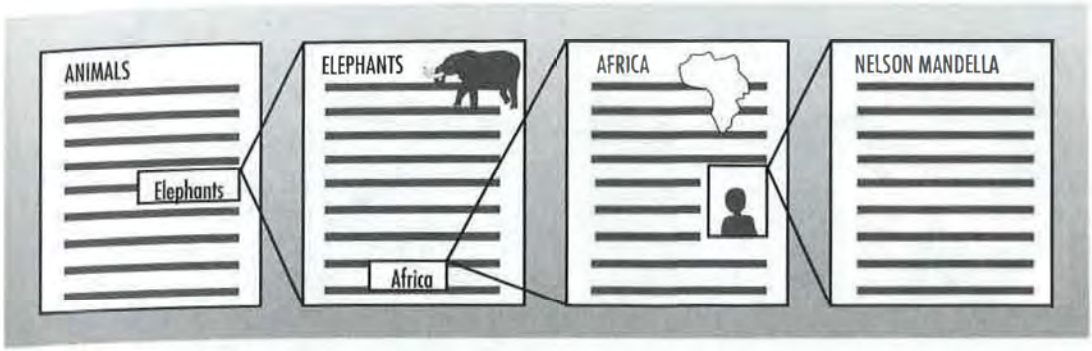


Figure 1–2 Hyper about hypermedia: a word or image on one page acts as the gateway to a different topic's page

Hypermedia Links

You're cozy in your favorite armchair reading a really good book about the Civil War when you come across a footnote that has something to do with the death of General Lee. You turn to the back of the book and find out that there's another complete book about General Lee's last days. You rush out to the bookstore and buy that book. It cites still other books. References like these let you keep on learning forever.

Hypertext is the same idea, computerized and made instant. You're sitting at your computer, reading about the Civil War, and you notice that the words "General Lee's death," are printed in blue. You click on the blue words with your mouse and an entire article about General Lee's last stand pops onto your screen. Hypertext combines documents by topics, not by sequence. That means you can jump from Tickling to Feathers to Birds to Penguins to The Arctic to Snow to Frosty-the-Snowman. Hypertext works the same way your mind does—-by association. Using hypertext you can access an infinite range of data, make sense of it faster than you ever have before, and have a lot more fun doing it.

The World Wide Web uses hypertext to link all its pages together. And since a page can be a movie, a graphic, a song, a database, an Internet resource, or a computer file, hypertext melds with multimedia to form *hypermedia*. See Figure 1–2 for an illustration of hypermedia.

A Web page may have no *hyperlinks* or hundreds of them. Sometimes a Web page is nothing more than a huge list of links, all geared toward one subject. There are two types of hyperlinks:

HTML Text links

HTML Graphic links

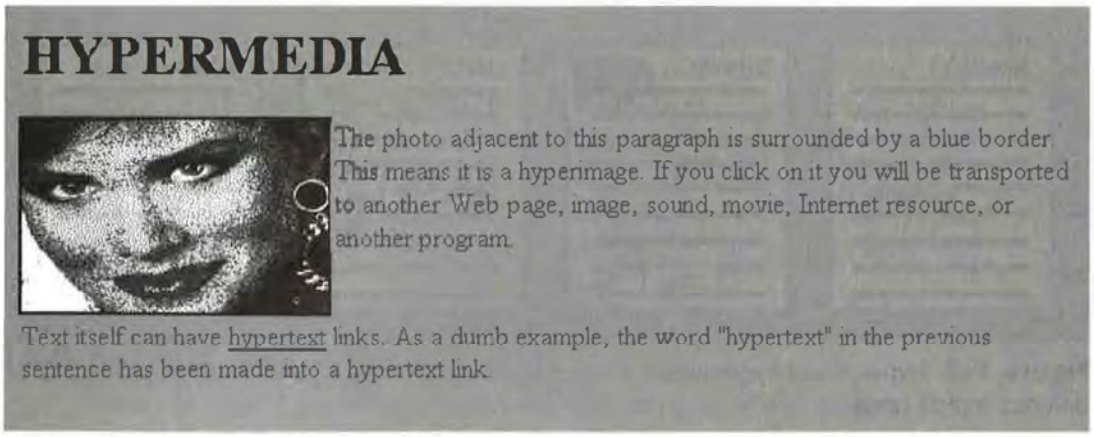


Figure 1–3 A typical Web page with various hyperlinks

A Web browser generally highlights any text link. For example, links in Mosaic and Netscape are generally underlined and colored blue, as shown in Figure 1–3. Cello surrounds the text with a perforated box. Lynx displays all links in bold print.

Graphic links may be small icons, buttons, logos, or any other photo, illustration, or design. A graphic link is usually surrounded by a blue border, also shown in Figure 1–3. To follow a graphic link, just click anywhere on the image. Some graphic links are *imagemaps*—clicking on different parts of the image transports you to different places. See the section on imagemaps in Chapter 12, Graphics, for more details.

You never know exactly where a hyperlink will take you, or exactly what will happen. Be brave, be whimsical, follow your gut. Most every browser has a Previous Page button or Back command. If you jump somewhere you don't really want to be, you can return to the previous page with the press of a key. Most browsers also have a *history list* that shows you all the pages you've visited and lets you return to any of them with ease.

Search Indexes

Hypertext makes it easy to get lost. That's part of the appeal. You can start out reading about Sperm Whales and end up reading about the Prince of Wales. But what if you want to find pages dealing with a specific topic?

Luckily, the Web is full of indexes. An index is a special Web page that allows you to type in a search term or keyword. The index then spits out a list of Web pages matching your search. You can then click on any page that

Figure 1–4 Searching for a Web starting point

catches your interest and transfer there automatically. Some indexes are manual: A person keeps track of all the Web pages he or she comes across. Others are created using *search robots*—small programs that crawl through the Web, following every hyperlink, keeping a list of every new place or topic they come across. Figure 1–4 shows a typical index search screen.

Many indexes allow you to search the full contents of thousands of Web pages. Other indexes let you search for titles, keywords, or geographical locations. Some indexes have a series of forms or menu items you can customize, allowing you to find exactly what you want. In most cases, you fill in a search term or phrase in a box and then click the Submit or Search button.

Some of the most popular indexes, and their URLs, are listed in Table 1–1. If you don't know what a URL is, check out URLs, URLs, URLs section, later in this chapter.

Table 1–1 Places to go Web hunting for that topic, file, or resource of interest

Search Index	URL
ALIWeb Archie Search	http://web.nexor.co.uk/aliweb/doc/aliweb.html
ArchiePlex Form File Search	http://www.lerc.nasa.gov/Doc/archieplex-httpd.html
CUI Index of Search Engines	http://cui_www.unige.ch/meta-index.html

Continued on next page

Continued from previous page

Search Index	URL
CUI Web Catalog	http://cui_www.unige.ch/w3catalog
CUSI Index of Search Engines	http://Web.nexor.co.uk/susi/cusi.html
Home Pages Broker	http://www.town.hall.org/brokers/www-home-pages/query.html
JumpStation Search Page	http://www.stir.ac.uk/jsbin/js
Lycos Search Engine	http://lycos.cs.cmu.edu
WebCrawler	http://webcrawler.cs.washington.edu/WebCrawler/WebQuery.html
WWWW: World Wide Web Worm	http://www.cs.colorado.edu/home/mcbryan/WWWW.html
Zorbamatic Title Search	http://www.rns.com/cgi-bin/nomad

THE HTTP PROTOCOL

When communication fails it's usually because of intelligence. Maybe the speaker is more intelligent than the listener, or vice versa. Or, more likely, two people are intelligent in very different ways—say, one is an expert in Swahili and the other speaks only Latin. Intelligence is a similar problem over the Internet. You can design a great way to ship data—fast, accurate, with lots of options. But unless other net surfers conform to your standard, they won't see your data.

The Web works so well because, at its heart, it's so stupid and trusting. The Web doesn't care what type of data you pump out, or what program you use to view the data. It treats everything the same. This means everybody can use their own browsers and their own data formats, and it'll all be friendly. You don't need to use different programs to access different types of data—the Web handles it all without complaining. All Web clients should be able to communicate with all Web servers.

The secret behind this is a simple, no-frills data protocol known as the *Hypertext Transfer Protocol* or *HTTP*. HTTP has the capacity to deal with most any Internet tool, file format, or resource. A Web browser client and the HTTP server communicate, coordinating things, before a Web page gets sent. If you could hear them talk, they'd sound something like this:

BROWSER: Hey, here's the page I want, here's how I want it.

HTTP SERVER: Uhh... sure. Can you handle JPEG graphics?

BROWSER: Yup. I sure can.

HTTP SERVER: Ah, good. Here you go, then. Here's your page, hot off the press.

NOTE: Web servers are usually called *httpd*, which stands for HTTP Daemon. A daemon is a program that hides in the computer's memory and is

always active. To learn all about servers, including how to set one up, flip to Chapter 21, Starting Your Own Web Site.

URLS, URLS, URLS

A *URL* (or *Universal Resource Locator*) is the address of a document, file, or other Internet resource. If you're familiar with the Internet, you know that each person has his or her own e-mail address. Each resource also has its own address. The World Wide Web gives every article, newsgroup, database, computer, file, or other Internet fragment its very own URL. A URL can consist of these elements:

- HTML The scheme, protocol, or type of Internet *service* the resource is using
- HTML The name of the *host* computer on which the resource is stored
- HTML The *port* number of the service
- HTML The directory where the resource is stored
- HTML The name of the resource itself
- HTML A term to search for in an index

Some of these attributes are optional, depending upon the type of resource. The port attribute, for example, is almost never used.

A URL will typically look like the phrase shown in Figure 1–5. In general, you don't need to be an expert at crafting or even decoding URLs. A browser always shows you the current document's URL, usually at the top of the page. Whenever you use your Web browser to move anywhere or retrieve any file, its URL is shown. It's a good idea, then, to brief yourself about the basic types of URLs so when you come across them, you'll know what to expect.

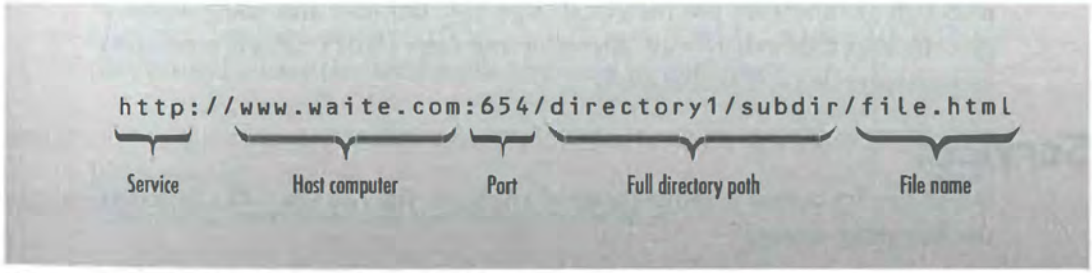


Figure 1–5 The breakdown of a typical URL

If you're reading a magazine, newspaper, or book and come across a URL that sounds pretty groovy, you can usually just type it into your Web browser. Your browser will then retrieve, display, or save the appropriate information on your hard disk.

NOTE: Since a URL usually accesses a Unix file system, upper and lower case are important. Be careful. Most URLs are expressed in lower case. Also, unlike DOS, Unix file and directory names can be any length. Don't be surprised when you come across a 20 letter filename. Another difference is directories: Unix doesn't use backslashes and drive-letter colons. The DOS file **C:\DIR1\DIR2\DIR3\FILE.HTM**, for example, is expressed in a URL as file:**///C|/dir1/dir2/dir3/file.htm**.

The Composition of a URL

The first part of the URL is always the name of a given service (see the next section for a list of services) followed by a colon. The second part of most URLs is the full Internet address of the machine where the resource is located. A host address always begins with a double slash: //. Most addresses are located at default ports, but in some cases you need to specify a specific port by following the address with a colon and then typing the port number. To specify the default port, simply leave out the colon.

After the host computer address, you need to specify the exact directory where the given resource is located. Type a slash between each level in the directory. Finally you can end the URL with the filename of the resource. This filename can be a hypertext document, a database, an index, a text file, a sound file, a graphic, or pretty much any other computerized resource in the world.

In some cases, the filename is followed by the hash mark (#) to specify a particular section within the file. A filename may also include a question mark (?) if it is an index or fill-in form, designating a term to be searched for. Search terms following a question mark are usually separated by the plus sign (+) and may use the equals sign (=). You may also come across a percent sign followed by two letters or numbers (%d1), which represents various extended ASCII characters.

Services

A URL always begins with the name of a service. You can use URLs to access the following services:

HTML http

HTML ftp

HTML file

HTML gopher

HTML mailto

HTML news

HTML telnet

HTML tn3270

HTML rlogin

HTML wais

New services are always being implemented as new Internet tools, protocols, and resources are created. If you come across a URL not listed here, try it out and see what happens.

Most of the time, you use URLs to call another hypertext Web page using the http service. You can also link documents to ftp file libraries, gopher menus, telnet machines, e-mail addresses, WAIS databases, or any other Internet resources. Chapter 16 discusses these resources in more detail.

Whenever you click on a hyperlink, you're actually requesting a specific URL. The Web server searches far and wide until it finds the URL, which it then delivers to your screen.

HTTP

The http service quickly and consistently calls up any Web page, text, graphic, or other supported item. Most Web pages are written using the *Hypertext Markup Language (HTML)*, which is covered in Chapter 10. HTML files generally have the .html extension, such as *index.html*.

To reference the file *index.html* in the */web/smith/* directory of a computer at address *www.smartpants.edu*, for example, you'd use the following URL:

```
http://www.smartpants.edu/web/smith/index.html
```

Most Web servers automatically search for a default file named *index.html* or *home.html*, depending on the brand of server and its current settings. If this is the case, you don't need to specify a filename to get an index. Just enter

```
http://www.smartpants.edu/web/smith/
```

Here are some sample valid—but fictional—URLs, ranging from the most basic to the most complicated:

HTML http://www.moolah.com/

HTML http://www.moolah.com/money/moremoney/

HTML http://www.moolah.com/money/moremoney/money.html

HTML http://www.testlabs.edu/Experiments-In-Science/Monkeys.html

HTML http://www.testlabs.edu/Experiments%2FIn%2FScience/Crocodiles

HTML http://strange.org:1234/strangness/hmm.html

HTML http://strange.org/strangness/hmm.html#novelty

HTML http://bigindex.edu.ca/reference/dictionary?cumquat

HTML http://abc.123.com:1234/abcdefg/hijk/lm/n/o/p/q/r/s/t/u/v/search?h*%20this%20now

FTP, File

A URL using the ftp or file service retrieves a file. To log into the FTP archive at Netcom, for example, you would use the URL:

```
ftp://ftp.netcom.com/
```

Most browsers then show you an index of files. Each file type may even be preceded by a special icon, letting you know whether it's a directory, a text file, an image, or whatnot. You can then click on various directories until you come across a file you're interested in. Just click on that file to download it onto your hard disk or display it on your screen.

NOTE: The Web server automatically logs you in as an anonymous user. Most public FTP sites happily accept anonymous visitors. You can, however, specify a specific username and password if you've got them; this may get you wider access to the site. See the FTP section in Chapter 16, Other Web Resources, for more information.

You can also use a URL to access a particular directory directly. For example, the URL

```
ftp://ftp.netcom.com/pub/pc/win3/games/
```

would automatically place you in the */pub/pc/win3/games/* directory, as long as such a directory exists.

You can even use a URL to immediately download a specific file. For example, if you wanted the file *pacboy.zip* in the */pub/pc/win3/games/* directory, you would use the following URL:

```
ftp://ftp.netcom.com/pub/pc/win3/games/pacboy.zip
```

The file service is almost identical to the ftp service, but is a quicker way of retrieving local files. You can use file URLs to load and display sound, image, or text files, as follows:

```
file://www.smartpants.edu/soundoff.au
```

Gopher

To connect to a gopher menu, simply use the gopher URL followed by the appropriate gopher code:

```
gopher://gopher.uchicago.edu/
```

Most browsers print the gopher menu on your screen, complete with different icons for different types of services.

You can use the gopher service to access everything available through ordinary gopher menus: phone books, search indexes like Veronica, images, sounds, or files. For example, to retrieve a specific phone-book index, your URL would look something like

```
gopher://mainman.herenorthere.edu:1234/phones.txt
```

Mailto

This is a relatively new URL that lets you send e-mail to a specific user. For example:

```
mailto:dsf4185@graphics.cs.nyu
```

Browsers which support the mailto URL automatically enter a text editor, allowing you to fill in a subject line, return address, and message. Your browser can then send your letter across the Web, quickly and easily.

News, NNTP

Some browsers use the news service to access your news server, retrieving an entire newsgroup, a specific message, or accessing anything using

Usenet's NNTP (*Network News Transfer Protocol*). To access a particular newsgroup, you'd use a URL similar to

```
news:alt.cream.cheese
```

You can even access a specific list of groups by using search marks like the wildcard asterisk or the question mark. For example, to see every group that starts with *alt.p*, you could use the URL:

```
news:alt.p*
```

Telnet, rlogin, tn3270

You can use the telnet, rlogin, or tn3270 services to log into a remote computer. Once you access one of these URLs your browser automatically dials up the remote system. You can use this to enter Internet-based bulletin boards, mainframes, MUD games, or other distant resources.

The telnet, rlogin, and tn3270 services are essentially the same. In general, you use telnet for most connections and tn3270 to telnet to 3270-type mainframes. A typical telnet URL would look like:

```
telnet://bbs.neato.com:1234
```

WAIS

The wais service links the Web to a *Wide Area Information Service* (*WAIS*) database or document. You can include specific search terms, databases, or other WAIS items.

ACCESSING THE WEB

You now know more about the history and theory of the Web than you really need, but you probably can't picture how everything fits together. The only way to grasp this unique medium is to experience it. Depending on where you live, the type of computer you have, and your technical expertise, connecting to the Web can range from a few minutes' work to a day or two of detailed software installation and setups. Don't sweat it too much, though. This book guides you through the steps. In fact, you may already be connected to the Web without knowing it.

What Computer Do I Need?

Most any computer worth its weight in salt can access the Web. That's part of the Web's appeal. However, this book focuses on PCs running Windows. If you have a choice, get a 386 or better PC with at least 4 MB of memory, a

hard disk with lots of free room, and Windows 3.1 or Windows for Workgroups. To take full advantage of this book, you need a CD-ROM drive as well.

You also want a Super VGA monitor, so Web graphics look their best. It's also a great plus if your PC has a sound card and speakers, allowing you to hear speech, music, or sound effects, all at CD quality.

What About the Modem?

If your computer is already connected to the Internet through a Local Area Network (LAN) or some other direct connection, then you won't need a modem. However, in most cases you need to call up a commercial Internet provider. A modem lets your computer communicate over the phone lines.

If you want graphics, sounds, and other huge files delivered to you in less than an eternity, you need a high-speed modem. Get one of these:

HTML V.32 9600 baud

HTML V.32bis 14,400 baud

HTML V.34 28,800 baud

The V.32, V.32bis and V.34 specifications are very important. These are standard protocols used by most Internet providers, so you want your modem to use them too. Modems using other protocols will usually transfer data just fine, but not at their advertised best speeds.

If you're just buying a modem, the V.32bis 14,400 is probably the most cost-effective choice, providing a good mix of speed and economy. You should also stick to a brand name; they don't cost much more than no-name brands, and they're less likely to have bugs or other errors to slow them down.

NOTE: If you want to use a 2400-baud or slower modem to access the Web, you should use a text-only browser like Lynx. This browser doesn't require a direct SLIP Internet connection: any Internet account will do.

What Type of Internet Account Do I Need?

You have several choices. The most convenient way of accessing the Web is to become part of a network. If you belong to a large corporation, organization, military group, or university, your institution probably already has Internet access. If your machine is connected to a Local Area Network (LAN), then all you may need is some software allowing your PC to

communicate via the Internet. Ask your system administrator if your personal PC has Internet connectivity. The system administrator can also help you set everything up.

If you're using a home computer, however, you need to use your modem to connect to the Internet. In most cases, the easiest way to do this is to call up a commercial Internet provider and use its computing power.

You may already be connected to the Internet through a standard *Unix shell* account. This means you dial up a service using a terminal emulator and can type commands like *telnet, ftp, gopher,* and *mail.* In most cases, this standard shell account is not powerful enough to connect efficiently to the Web. Instead, you need to make your home computer an actual part of the vast network. This requires network-level access, which is known as SLIP or PPP. However, there are several options available to the standard Internet user.

The next few sections cover these methods of hooking up to the Web:

- HTML SLIP and PPP accounts
- HTML Standard Internet Unix shell accounts running a text-only browser
- HTML Standard Internet accounts running the SlipKnot browser, a full graphical browser that runs hand-in-hand with Lynx
- HTML Standard Internet accounts running The Internet Adapter, a complete SLIP emulator for the Unix shell

SLIP ACCOUNTS

SLIP has nothing to do with banana peels. Rather, it stands for *Serial Line Internet Protocol. CSLIP* is compressed SLIP, a variation typically used over the telephone. *PPP* is the *Point to Point Protocol.* All three protocols do the same thing for you: they let your computer connect directly to the Internet through phone lines instead of through network wires.

If you have a regular old Unix shell, your computer acts only as a window for another large computer, somewhere up in some office building, which is attached to the Internet. A SLIP account skips the middleman—your personal machine actually becomes another corner of the Internet (see Figure 1–6). This means you can easily and conveniently use any PC Internet client, including Mosaic, Netscape, or any other Web browser that comes along. You can also use a SLIP account with cool graphical Windows Internet programs like WinGopher, the Trumpet Winsock newsreader, or

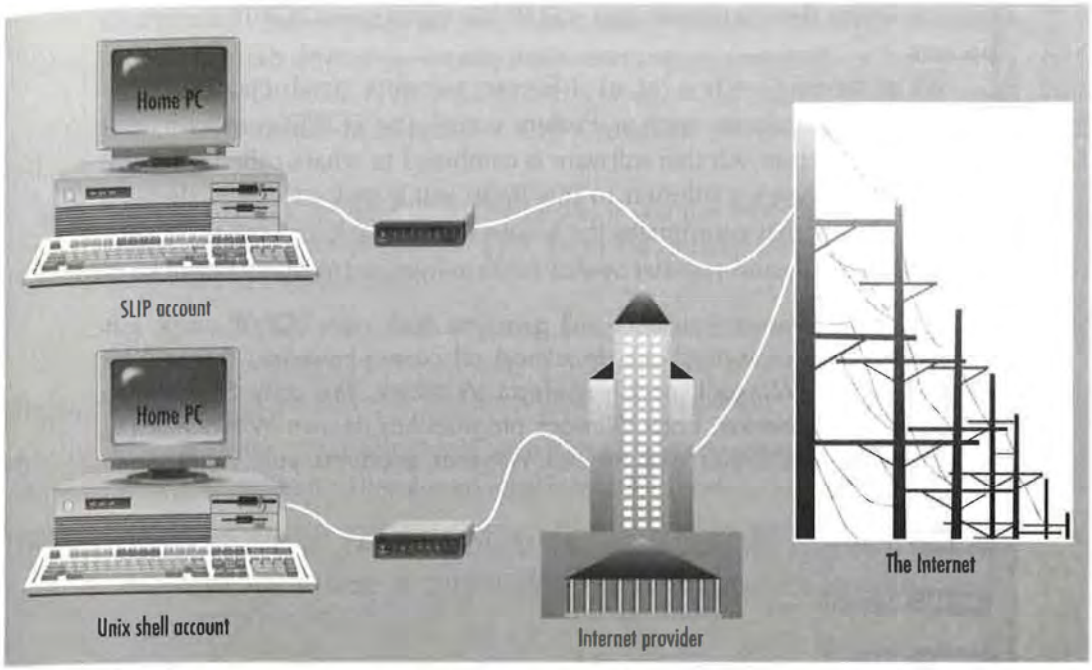


Figure 1-6 A SLIP account compared to a standard Unix shell account

Eudora e-mail. These shareware products are discussed in Appendix B. SLIP access is also generally faster than a Unix shell.

Most every major Internet provider is beginning to offer SLIP or PPP accounts. Some of the most popular are listed in Appendix A, Internet Providers. SLIP often costs a few dollars more than a standard Unix shell account, though prices are steadily decreasing. You should be able to find an unlimited SLIP account for about $10 to $20 per month.

TCP/IP

SLIP itself just connects your machine to the rest of the Internet. But just as a telephone line is worthless without a telephone, a SLIP account is worthless if it can't carry data back and forth. To do anything useful with SLIP, you need to speak the same language as the rest of the Internet: the *TCP/IP*, *Transmission Control Protocol/Internet Protocol*.

The Internet is comprised of millions of machines worldwide. TCP/IP is the standard they all use to talk to each other. Every Internet client or Web browser speaks using TCP/IP *packets*—small bundles of computerese. Before you can do anything with your account, you'll need some pretty

smart software that can speak over a SLIP line using these TCP/IP command packets.

We're dealing with a lot of different software products: The Web browser, Internet clients such as Eudora e-mail, the TCP/IP translator, and the SLIP connection. All this software is combined in what's called a TCP/IP *stack*. Luckily, there's a solution to this mess: put a sock in it. Yes, there's a type of software that coordinates the whole TCP/IP stack for you: the *Socket*. The most versatile and popular socket for Windows is Trumpet Winsock.

NOTE: Many Internet providers will give you their own TCP/IP stack software, with its own instructions. In almost all cases, however, this will be either Trumpet Winsock or Chameleon Winsock, the only Shareware Winsocks on the market. Each Winsock program has its own *WINSOCK.DLL* Windows library. Other commercial Winsock products you might come across include:

- HTML Spry
- HTML Novell
- HTML FTP
- HTML NetManage
- HTML PC/NFS

Installing and Configuring Trumpet Winsock

Trumpet Winsock can be found at the site listed in Appendix D. To install everything, just unzip and copy the entire directory to your hard disk. You may want to create a directory similar to

```
C:\WINSOCK
```

You now have a choice. If you're familiar with DOS, you can add this directory to your path by changing your AUTOEXEC.BAT file. For example, if your current path is

```
PATH C:\WINDOWS;C:\DOS;C:\UTIL
```

then you add your Winsock directory to the end of the line, after a semicolon:

```
PATH C:\WINDOWS;C:\DOS;C:\UTIL;C:\WINSOCK
```

You should now reboot your computer to make the new path take effect.

If you prefer, you can just move the *WINSOCK.DLL* file to your C:\WINDOWS directory—or any other directory in your path.

You should also set up an icon for WinSock's *TCPMAN.EXE* file in your Windows Program Manager, since it's a program you'll be accessing quite often.

NOTE: Trumpet Winsock is not freeware. If you use Winsock regularly, you should register your version for $20. Read the registration information in the included INSTALL document.

You now need to set up Winsock so it knows how to communicate. At this point, you should already have a SLIP account at an Internet provider. Your provider should give you the following information:

HTML Phone. The telephone number of your Internet provider's SLIP computers. This should ideally be a local call.

HTML Username. The name you'll use to log in to the system; it may be your last name, a combination of your initials, or a meaningless ID number.

HTML Password. The code you'll use to assure the system that you're the one using your account. It can be any series of letters or numbers. Keep track of upper and lower-case letters—if your password is Froggie, "froggie" may not get you into the system.

HTML IP Address. Your provider's Internet Protocol address, in the IP number form 1.2.3.4. The value for this may be the word *bootp* (in lower case) or the value 0.0.0.0.

HTML Name Server. The IP number of the machine you'll be calling to look up Internet domain names.

HTML Gateway. The address of the Internet gateway or router being used.

HTML Time Server. The IP number of the Time server machine. Usually unused or the same as the Name server.

HTML Domain Suffix. The address of your Internet provider, in the form *name.com*. You may need several of these.

The first time you run Winsock, the Network Configuration window appears, as in Figure 1–7. To return to this window in the future, select File, Setup.

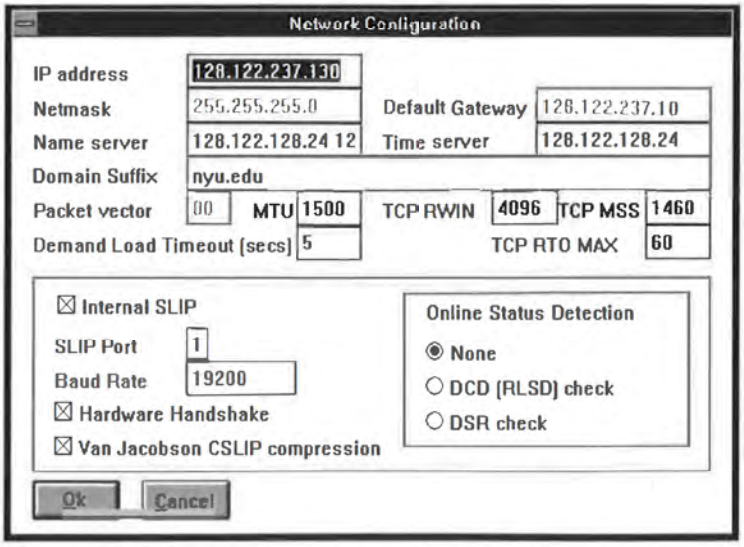


Figure 1–7 Configuring Trumpet Winsock

Check the Internal SLIP check box. Type the COM port your modem is connected to in the SLIP Port box. This should be a number from 1 to 4. For example, if your modem is wired to COM2, type 2. Fill in your modem's maximum baud rate. Do not use commas as part of the number. If you're running 14,400 baud you should fill in the maximum data rate:

19200

Select the Hardware Handshake checkbox unless your Internet provider tells you not to. Typically, you need to check the Van Jacobson CSLIP Compression box too. This means you'll be using CSLIP instead of SLIP, which is a common way of sending SLIP over telephone lines. If you know your modem can handle DCD (RLSD) or DSR Online Status Detection, you can click one of these values. Otherwise, click None.

You can now fill in the IP Address, Name Server, and other values. If you're supposed to use several Domain Suffixes, separate each by a space. You now need to fill in some specific data speed values. The values for these depend on whether you're using SLIP or CSLIP compression. If your Internet provider doesn't give you exact values, try the defaults:

HTML MTU. This is the Maximum Transmission Unit, which depends on the Maximum Segment Size (MSS). Winsock recommends you use the MSS value plus 40.

HTML TCP RWIN. The TCP Receive Window. Winsock recommends you use your MSS number times 3 or 4.

HTML TCP MSS. The TCP Maximum Segment Size. This number should be around 512 for SLIP or around 255 or less for CSLIP.

You should leave all other values at their defaults, unless your Internet provider tells you otherwise. When everything is filled in, click OK. A message appears, letting you know that you must restart Trumpet Winsock before any of your settings will take effect. Exit Winsock by selecting File, Exit and then reload it.

NOTE: If your machine is connected directly to the Internet through a Local Area Network, you need to install a TCP/IP packet driver such as WINPKT. You can then use SlipKnot to handle the stack. Internal SLIP should be unchecked. Ask your system administrator for the value of your institution's IP Address, NetMask, Default Gateway, and all other values.

Manual Login

You now need to call your Internet service provider to be sure everything works. At first, make the call manually. Eventually, you want to create a script file that can automatically dial the Internet provider, log in using all appropriate names and passwords, and start up your SLIP connection using the proper commands.

To use the manual login, select Dialler, Manual Login. The following message should appear:

```
Manually dialing.
AFTER LOGGING IN, PRESS THE <ESC> KEY TO RETURN TO NORMAL SLIP
PROCESSING
SLIP DISABLED
```

You now need to use your modem's built-in dialing commands to call up your Internet provider. The dial command is ATDT followed by a valid phone number. For example, if your Internet provider is located at 555-1234, you would type

```
ATDT 555-1234
```

The Winsock program then becomes a generic terminal-type communications program. What you type next depends on your Internet provider. The provider should give you full instructions. In some cases, you may need to type the word *SLIP*, or just press [ENTER]. In any case, you should eventually see a login prompt similar to

```
login:
```

Type your username and press [ENTER]. You should now be asked for your

```
Password:
```

Type in the password. In most cases, you need to start the SLIP protocol by typing the word

```
slip
```

A message similar to the following should now appear:

```
Switching to SLIP.
Annex address is 123.456.789.10. Your address is 123.456.7.8.
```

What you do next depends on whether you have a *static* SLIP account or a *dynamic* one. If your Internet provider assigned you your own personal IP number then you have a static account and everything should be set up fine. You should now run the *bootp* protocol, to get specific addresses, ports, and other information from your network. In most cases, you need only type:

```
BOOTP
```

Press [ESC] to stop the manual login process. The following message should appear:

```
SLIP ENABLED
```

However, many Internet providers use dynamic accounts instead of static ones. These providers have several valid IP numbers, and assign them on a per-call basis. When you dial in on a dynamic account, you get one of the IP numbers no one else happens to be using at the moment. To finish the login process, you need to select File, Setup and type your assigned IP number in the IP Address box. Click OK. A message appears, telling you that you need to close Winsock before any change can take effect. Press the [ESC] key, exit Winsock (by selecting File, Exit), and then restart Winsock.

You can now run Mosaic, Netscape, Eudora, FTP, or any other whizbang Internet software. If things don't seem to be working, check out the Shooting Down Troubles section later in this chapter. Otherwise, you're ready to set up an automatic script.

Login Scripting

If your manual login seemed to work just fine, it should be easy to create an autopilot that dials you up, logs you in, and sets everything running. Many Internet providers can give you a customized Winsock script file if you ask. The file should be called *login.cmd*. Copy the file to your Winsock directory. To log in now, just select Dialler, Login.

NOTE: You can have Winsock automatically call your Internet provider every time you run it. Select Dialler, Options. Click the Automatic Login On Startup Only radio button and select OK.

There are several useful script commands you need to know:

HTML *input <timeout> <message>*. Waits <timeout> seconds for the <message> to appear.

HTML *output <message>*. Sends the <message> over the modem.

HTML *password <prompt>*. Open a dialog box asking the user for the password.

HTML *username <prompt>*. Open a dialog box asking the user for a username.

HTML *BOOTP*. Run the bootp protocol.

HTML #. A comment.

HTML \13. Send a carriage return; the same as pressing the ENTER key.

HTML \i. Send the specified IP address.

HTML \p. Send the password that was obtained in the Password dialog box.

HTML \u. Send the username that was obtained in the Username dialog box.

The idea is to write a small program that mimics the exact steps you performed during the manual login. Every script begins with the ATDT dialing command. You should then wait for a specific prompt. For example, if the first thing your Internet provider requires you to do is type *login* at the *Welcome>>* prompt, the beginning of your script would look like:

```
#Call up your Internet provider
output atdt555-1234\13
#Wait for the CONNECT message
input 30 CONNECT
#Send <ENTER> to kick the system into action
output \13
#Wait for the Welcome>> prompt; you only need to specify the last
#few letters.
input 30 come>>
#Type the login command
output login\13
```

The best way to begin writing a script is to edit Winsock's sample login script and fill in the proper phone number or connect values. Select Dialler, Edit Scripts. The File Open dialog box appears. Select the *login.cmd* program and choose OK. A sample script follows. You should replace the sections in boldface print with valid values.

```
# Dial-in script to a typical SLIP Internet provider
#
#                       Initialize modem:
output at\13
#               Wait for 'OK' from modem
input 10 OK\n
#               Dial the Internet provider's phone number
output atdt555-1234\13
#               Wait for the CONNECT message
input 30 CONNECT
#               Wait for Username: prompt
input 10 name:
#               Prompt user for username using a dialog box
username Username
#               Type out that username
output \u\13
#               Wait for Password: prompt
input 10 word:
#               Prompt user for password using a dialog box
password Password
#               Type out that password
output \p\13
#               Invoke the SLIP protocol by typing the word "slip"
#                       You may have to move this line prior to the
#                       logging-in step.
output slip\13
#               Wait for IP address string
input 30 Your address is
#               Parse address (for dynamic SLIP accounts)
#                       If you have a static slip account you can
#                       delete the next line
address 30
input 30 \n
display \n
display Connection Attempt Complete!
display \n
#               Get network information using BOOTP protocol
BOOTP
```

Logging Out

To hang up manually, select Dialler, Manual Login. Type three plusses:

```
+++
```

and then type the modem hang-up command:

```
ATH
```

You can also log out using the standard logout script. Select Dialler, Bye.

UNIX SHELL WEB BROWSERS

A Unix shell is the traditional way of accessing the Internet from a home computer. You dial up an Internet provider's modem and, using a terminal program, you control the provider's computer and talk through it to the Internet.

This gives you command of the Internet, but it has many limitations. Since you're using somebody else's computer, you are limited to a simple form of communication. In most cases, you can only view or type text. Anything like graphics gets far too complicated.

A SLIP account actually makes your home computer a part of the Internet. But sometimes even SLIP isn't the answer. For instance:

- HTML Your modem may be too slow; less than 9600 baud
- HTML SLIP may be too expensive in your area
- HTML There may be no local SLIP provider
- HTML Your computer may not have the resources to run SLIP
- HTML You're already connected to the Internet using a great Unix shell and don't want to give this up

Just because you're limited to a Unix shell account doesn't mean the situation is hopeless. Far from it. There are several surefire ways to access the Web from your Unix shell. The easiest way is to use a Unix-shell-compatible Web browser, such as the text-only Lynx or WWW or a graphical browser such as SlipKnot.

Text: Lynx or WWW

The simplest solution is to get a text-only browser like Lynx or WWW. These browsers run on your Internet provider's Unix machines, not on your home PC. Most Unix servers, however, should already have a working copy ready for you to use. Lynx and WWW allow you full access to the Web with one catch: You won't be able to see any graphics, hear any sounds, or be wowed by any other multimedia perks. The Lynx browser is covered fully in Chapter 3. If you can't run a local version of Lynx or WWW, you can telnet to it. See the Telneting To A Public Browser section in the next chapter.

To access Lynx or WWW, you can usually just go to your Unix prompt and type

```
lynx
```

or

```
www
```

Graphics: SlipKnot

MicroMind's SlipKnot is a terribly clever graphical Windows Web browser that works using dial-up Unix shell accounts. You need no SLIP, PPP, or TCP/IP software of any kind. SlipKnot loads up Web pages much more sluggishly than a SLIP Web browser, but if your modem is fast enough it's a very viable option. SlipKnot also has a number of other nice features. For instance, once you load up a Web page, you can load it again instantaneously. You can also view several Web pages at once.

SlipKnot is covered fully in Chapter 4.

THE INTERNET ADAPTER

If you can't access a SLIP account, the best solution is to emulate SLIP over your Unix shell. For $25, The Internet Adapter (TIA) software by Cyberspace Development, Inc., does this task most excellently. If Winsock is the socket, TIA is the three-pronged adapter that plugs the Internet into your home computer. Once TIA is installed, you can use Mosaic, Netscape, or any other multimedia Internet tools, the exact same way you would over a true SLIP line. A Unix Internet account equipped with TIA usually runs just as fast as any other SLIP.

Since TIA rides atop your standard Unix shell, it's a good idea to be fairly familiar with basic Unix and Internet commands before you try installing it. You might want to check out a command-line Internet book such as the Waite Group's *Internet How-To*.

NOTE: If a SLIP account costs much more per month than a standard Unix shell account, The Internet Adapter often pays for itself quickly.

Determining Your Unix Platform

The Internet Adapter does not run on your home PC—it is Unix software and is stored and accessed on your Internet provider's hard disk. Before you install TIA, be sure you have enough space. If not, you may have to request

or buy additional space from your Internet service provider. Also beware: Some Internet providers do not allow you to use TIA because it interferes with their regular SLIP service.

All Unixes are not built the same. There are many platforms, or flavors, of the Unix operating system. Likewise, there are many different versions of TIA depending on which Unix shell you are using. TIA supports the following hardware/operating system configurations:

HTML Sun Sparc: Solaris 2.x or SunOS 4.1.x

HTML 386/486: BSDI 1.x, SCO, or LINUX

HTML DEC Alpha: OSF/1 2.0

HTML DEC: ULTRIX 4.3

HTML IBM RS/6000: AIX 3.2

HTML SGI: IRIX 4.0

HTML HP 9000: HPUX 9.0

If you're not sure what system you're using, you can ask your Internet provider. Alternatively, you can run a special script program to help you guess your brand of Unix. This script is not PC software. You'll need to use your Unix shell account to download the script directly via FTP. To log in to the FTP archive, type

```
ftp marketplace.com
```

When asked for a login name, type

```
anonymous
```

When asked for your password, type your e-mail address. Now type the following sequence of commands at the *FTP>* prompt to grab the file:

```
cd tia
ascii
get config.guess
bye
```

To make the script executable (able to be run), type

```
chmod u+x config.guess
```

And then type the following to run the script:

```
./config.guess
```

If worst comes to worst, you can download each version of TIA and try them out until you find one that seems to work.

Getting Your License Code

Since there are several different versions of TIA, you need to download TIA directly from its creators. Cyberspace Development issues free evaluation copies of TIA that you can try for 14 days. After that point, the software is automatically disabled.

To get a copy of the evaluation software, you need to order it, along with your own personal license code. To snag the order form, send a blank e-mail letter to: *tia-single@marketplace.com*. With most Unix accounts, you would type:

```
mail tia-single@marketplace.com
```

You could then press [ENTER] when asked for a subject and press [CTRL]-[D] to send the blank letter.

Cyberspace Development mails a user form to you immediately. When you get the form, load it up using any text editor and fill it in. You should fill in an answer to each question between the dotted lines and after the colon. For example, when it says:

```
-----------------------------------------|
# FIRST NAME:
-----------------------------------------|
```

you would put the cursor after the colon and type

```
-----------------------------------------|
# FIRST NAME: David
-----------------------------------------|
```

assuming, of course, your first name actually was David. Be sure to put an *X* next to the Evaluation order type when asked, unless you'd like to use your credit card and order a registered copy of the software right away.

```
-----------------------------------------|
# ORDER TYPE: Evaluation (X) Purchase ( )
-----------------------------------------|
```

When you're done, mail back the completed form to *tia-single@marketplace.com*. If you have any questions or problems with this process, you can send mail to:

```
tia-order-desk@marketplace.com
```

NOTE: After evaluating TIA for 14 days you may decide to buy a registered license code. You can resubmit the same form, this time including your credit

card or other payment details. You can also order TIA by phone using a credit card by calling SoftAware at 310-314-1466.

Shortly, you get two messages back: your License Code and Installation Instructions.

Getting the Software

Once you receive your TIA Evaluation License Code, save the e-mail message in a file in your *home directory* called *.tia*. You can do this using the extract command of your Unix mail program. Read the e-mail message that contains your TIACODE and then type:

```
s .tia
```

Alternatively, you can create the .tia file in your current directory using your favorite Unix text editor. Print or write down your TIACODE, Login Name, and Domain Name. You can then create a *.tia* file that looks similar to the following:

```
TIACODE:abcd22ggdg2h3g4yutfgfj2h3j4h3jh3jh2j3h3
Login Name: <jsmith>
Domain Name: <smartpants.edu>
```

NOTE: If you purchase a registered TIACODE, the process is the same. Be sure to put the latest code in your *.tia* file. You should also keep a copy of the code in a separate file or on your home PC so that if your Unix provider's hard disk crashes, you can easily reinstall TIA.

You're now ready to download the TIA program itself. You'll need to know what Unix platform you're working on. To access the FTP archives, type the following command at your Unix prompt:

```
ftp marketplace.com
```

When asked for your login name, type

```
anonymous
```

When prompted for a password, type your complete e-mail address. You can now access the TIA file by changing to the proper directory. Change to the *tia* directory by typing

```
cd tia
```

NOTE: Some versions of the software may be available in the *tiabeta* directory.

To see the available versions, type

```
dir
```

You can now retrieve the proper version of TIA by using the *get* command. For example, if you need the *sparc.solaris.tia* version, you would type the following series of commands:

```
binary
get sparc.solaris.tia
bye
```

Installing The Internet Adapter

You should now have a working copy of TIA in your Unix machine's home directory. You can rename the file you just retrieved to *tia* using the *mv* command. For example:

```
mv sparc.solaris.tia tia
```

You now need to make the TIA software executable by using the *chmod* command:

```
chmod u+x tia
```

You can verify that The Internet Adapter is working by typing:

```
./tia
```

The following message should appear:

```
The Internet Adapter (tm) 1.0 for BSDI Unix
Copyright (c) 1994 Cyberspace Development, Inc.
Single License
Ready to start your SLIP software.
```

WARNING: An evaluation copy of TIA only works for 14 days from date of issue, so it needs to get the current date from your Unix machine. The evaluation version will not run at all if it can't read the date. Unfortunately, some Internet providers do not allow public access to the time utilities. If you hit on this problem, you can still try TIA if you can persuade your Internet provider to allow time-server access, or if you go ahead and buy a registered copy of TIA.

Configuring Winsock for The Internet Adapter

Once The Internet Adapter is successfully installed on your Unix system, you can run the Trumpet Winsock program as usual. Instead of dialing a SLIP account, you dial up your Unix shell account and run The Internet Adapter. Before you make the call, however, you should fine-tune some of Winsock's settings so that it works in tandem with TIA.

The Setup

To set up Winsock, run it and select File, Setup. Be sure the Internal SLIP box is checked. You now need to fill in a number of values. Since TIA only

emulates a SLIP account, there is no actual IP address. Instead, you should use the dummy IP address of 192.0.2.1. Your Netmask should be 255.255.255.0.

The TIA program itself can figure out what value you need to put in the Name Server box. Dial up your Unix account and type:

```
./tia -address
```

TIA tells you the IP address for the Gateway and the Name server. If your machine has several Name servers, separate each with a space. If this command doesn't work, you can try Unix's own Name server command:

```
whois -h internic.net domain.xxx
```

If neither of the above two commands seems to work, you'll need to ask your Internet provider for the IP number of the Name server.

You can leave the Time Server and Domain Suffix boxes blank. You now need to set the data flow values. Use the following values:

HTML MTU: 1500

HTML TCP RWIN: 4096

HTML TCP MSS: 1460

HTML TCP RTO MAX: 60

HTML Demand Load Timeout: 5

You should now fill in your Winsock communications settings, as described earlier in this chapter. Fill in the correct SLIP Port, Baud Rate, and Hardware Handshake values. Be sure the Van Jacobson CSLIP Compression attribute is *not* checked.

Trumpet Winsock should now be a perfect match with The Internet Adapter. Click OK, exit Trumpet, and restart it.

Dialing Manually

You can now use Trumpet Winsock to dial your Unix system manually, as described earlier in this chapter. Select Dialler, Manual Login. Dial up your system and log in the way your normally would. Instead of typing *slip*, however, you should type:

```
./tia
```

You should now see the message:

```
Ready to start your SLIP software
```

Press the ESC key to end the manual login sequence. Hearty congratulations to you! You now have full SLIP access to the Internet.

NOTE: If your Unix Internet account uses a series of menus, you need to find some way of getting to the Unix command prompt. Ask your Internet provider if a standard shell account is available. If not, you'll need to create a Winsock script to navigate the proper menu choices.

Dialing with a Script

Alternatively, you can create a Winsock script that logs into your Unix shell account and runs the TIA program. This script should look like the one earlier in this chapter, with the additional line:

```
output ./tia\13
```

Exiting TIA

To quit TIA manually, select the Manual Login item from Winsock's Dialler menu. Slowly type CTRL-C five times to exit the TIA program. You'll return to the Unix prompt, where you can use Unix Internet commands as usual. You can rerun TIA at any time.

SHOOTING DOWN TROUBLES

Stacks can often tumble. The complex combinations of TCP/IP software, socket software, SLIP software, Internet client software, Web browsers, your Internet provider's Unix system, your machine, and Windows itself can cause lots of headaches.

Whenever you have a problem, be sure to check each component individually. If you're using The Internet Adapter, run it and be sure it doesn't give you any error messages. If it does, you may need a different version of the software. If you're running a Winsock script that doesn't seem to work, first try logging in manually to see if the problem is with your script or with Winsock itself. Make sure your modem is connected when you think it should be: just pick up the phone and listen; if you hear a low-pitched grumbling noise, you're online.

That said, here are a few common errors and easy ways to squash them:

Your Modem Won't Respond

If you try dialing using Winsock but nothing seems to be happening, this could indicate one of several problems:

HTML Your modem isn't attached to your computer. Check to be sure the cables are inserted tightly.

HTML Your modem isn't attached to your phone line. Plug a phone into the line to be sure you're getting a dial tone.

HTML Your modem isn't turned on.

HTML Your modem hasn't been properly reset or initialized. Check your modem owner's manual for guidelines. If possible, contact your modem's manufacturer.

HTML Your port isn't configured properly in Windows. Run the Windows Control Panel and select the Port icon. Select the port your modem is connected to and click the Settings button. Be sure the settings correspond to your modem's baud rate. Note that a 14,400 baud modem's port should be set to 19,200. The port should also be set for 8 data bits, no parity, 1 stop bit, and the Flow Control type should be Hardware.

If your modem works fine one moment but then suddenly stops responding, the modem might just be a little tipsy. Turn off the modem and turn it back on or reboot your computer.

The "DNS Lookup Failed" Message

This is probably the most common error message you're likely to run across. It means your Web browser is having trouble accessing remote URLs.

Most likely, there's a wrong value in your Winsock setup window. Double-check the IP numbers of your Domain server and your Name server. Contact your Internet provider to be sure they gave you the right information in the first place. All these numbers can get confusing; you may have copied something down wrong. Using the wrong IP number is as useless as dialing the wrong phone number.

The "Unable to Resolve Host Name" Message

If everything seems to run, but your Internet tools have problems connecting to specific hosts, your Domain Name Server (DNS) information is probably wrong. Check SlipKnot's setup and double-check the Name server entry.

WHAT NOW?

You're now connected to the Internet, either using SLIP or The Internet Adapter. Your next step is to install, configure, and learn to use a Web browser so you can start flipping through those tens of thousands of multimedia Web pages. Chapter 2 introduces you to the wild world of the Web browser. Later chapters deal with particular brands of browsers:

HTML Chapter 3 is a hands-on guide to using the text-only Lynx browser.

HTML Chapter 4 covers the SlipKnot graphical Unix shell browser.

HTML Chapter 5 shows you everything you need to know about the acclaimed NCSA Mosaic browser.

HTML Chapter 6 guides you through the sleek Netscape browser.

HTML Chapter 7 takes a brief look at other Web browsers available for the Windows PC.

THE WAITE GROUP®

HTML WEB PUBLISHER'S CONSTRUCTION KIT

PUBLISHING YOUR OWN HTML PAGES ON THE INTERNET

HTML Web Publisher's Construction Kit provides everything the aspiring Web publisher will need to set up shop in the world's cutting-edge emporium of ideas, sound, and imagery.

For the novice, the book covers every detail, from obtaining the optimal Internet account to installing the converters, viewers, protocol transfer utilities, and communication sockets necessary for full Web capabilities. Accessible chapters walk the reader through the premium Web Browsers—SlipKnot, NetManage Chameleon—included on the enclosed CD-ROM.

HTML Web Publisher's Construction Kit explains in straightforward terms popular methods and options for creating Web pages. The book contains actual Web documents that are dissected, illustrating precisely how each aspect was created. Finally, it shows you how to upload and post your finished Web publications, providing a roster of free and commercial Web servers. *HTML Web Publisher's Construction Kit* is your road map to the exciting terrain of World Wide Web publishing.

- ✔ **Connect to the World Wide Web.** Access a world of information with a click of the mouse.
- ✔ **Browse the WWW.** Cruise easily with Lynx, SlipKnot, NCSA Mosaic, WinWeb, and tools Trumpet Winsock and The Internet Adapter.
- ✔ **Author WWW Pages.** Check out HTML specification 3.0, as well as authoring tools such as HTML Assistant and HoTMetal.
- ✔ **Design Top Documents.** Learn layout basics, CGI scripts, and Gopher, Usenet, FTP, e-mail, and WAIS display
- ✔ **Publish Without Perishing.** Upload and post completed Web pages, locally and via FTP.
- ✔ **Configure Killer Web Sites.** Construct impressive Web sites and servers under Windows, Unix, and Mac. It's easy!

Platform: PC, Mac, Unix
Operating System: Windows, MacOS, Unix
Requires: 386 DX or better; VGA card and display; 4 MB RAM; hard drive; CD-ROM drive; access to the Internet

$36.95 USA
$53.95 Canada

SHELVING: INTERNET

ISBN 1-57169-018-2

53695>

9 781571 690180

WAITE GROUP PRESS™
200 Tamal Plaza
Corte Madera, CA 94925

LIBRARY OF CONGRESS
0 051 394 291 6