HAHN

INTERNET
COMPLETE
REFERENCE

LIBRARY OF CONGRESS
0 003 563 116 1

FLM
2015
232009

TK 5105
.875
.I57 H34
1996

THE LIBRARY OF CONGRESS

OSBORNE

HARLEY HAHN
ONE Million
MORE THAN ONE MILLION BOOKS SOLD

The Internet Complete Reference

Second Edition

By the Author
of the Best-Selling
Internet Yellow Pages

"...the best how-to book is the 'The Internet Complete Reference'..."
The Washington Post

"...incredibly thorough..."
The Wall Street Journal

Harley Hahn

Meta's Exhibit 1008
Page 003

A PERSONAL NOTE FROM HARLEY HAHN

The Internet is, by far, the greatest and most significant achievement in the history of mankind. What? Am I saying that the Internet is more impressive than the pyramids? More beautiful than Michelangelo's David? More important to mankind than the wondrous inventions of the industrial revolution?

Yes, yes and yes.

Do I expect you to believe this? Of course not—not right now anyway. However, for years now, people have been connecting computers into networks, and now we have the Internet: a worldwide network connecting millions of computers and millions of people. What is amazing is that, within a few short years, the Internet has changed our civilization permanently and has introduced us to two completely unexpected ideas.

First, hundreds of thousands of people have been laboring to build the Internet. They have worked alone, in small groups, and within organizations, like so many ants in a global anthill. Many of these people are only doing their jobs but, just like the ants, they serve the common good while having no conception of the order and of the compelling forces that drive their work.

Deep inside us, there is an irresistible genetic program to follow. There is a voice that we hear only subconsciously and only as a species. A voice that commands us to take our computers, connect them into networks and... communicate.

This leads to the second great idea: When we connect computers we invariably create something that is well beyond the sum of its parts. Although there are millions of computers on the Internet, the computers are important only in that we use them to follow our instructions and to store our data. What is more important is something we cannot yet understand: the answer to the question, what happens when millions of people gather in a safe place to talk and to share? That is what the Internet is about, and that is what we are just beginning to find out.

I sense we are near the beginning of a great and important change in human affairs. There is a reason we built the Internet and, as human beings, we have an obligation to learn how to use it and to participate.

Now, about this book.

Before we start—you and I—I want to make sure you understand an important point: the Internet is easy to use, but is difficult to use well. I have filled this book with technical details, important details you must understand and master in order to use and enjoy the Internet. Still, don't for a moment think that you need to be a computer expert. The Internet is not for nerds, but just as surely it is not for dummies: it is for those people who are willing to think and to learn and to contribute.

If you ever have one of those days where everything seems to go wrong, and the computer just won't cooperate, and you can't understand what is happening, take a moment and try to see things in a larger sense. As a human being, you share the birthright of intelligence, curiosity and, above all, the ability to learn. However, these gifts are not free. By your very nature, you can not only learn, you are compelled to learn if you are to remain happy and fulfilled. Still, as the saying goes, there is no royal road to knowledge.

To use the Internet, you will have to expend some time and some effort (actually, a great deal of time and effort). But if you do, here is what I promise you:

- I promise I will stay with you.
- I promise I will guide you.
- I promise I will teach you the new words and the new ideas and the new skills you need to use the Internet well.

I have taken a great deal of time to organize this book so as to lead you from one topic to another in a way that will make sense to you (at least, in retrospect). However, you do not need to read everything from start to finish (although I think you should). Actually, this book comes with only two simple instructions:

1. Start by reading chapter 1.
2. When in doubt, do something enjoyable that requires effort.

I know what it is like to be faced with a large, inconceivable mass of unknowable information. What you are holding in your hand is much more than a computer book. It is a connection—a real connection—between you and me. Your job is to put in the effort and the time. My job is to open the doors.

I wrote this book to be your companion and, I promise you, it will be a good one. Take a few moments and skim through the pages. This is not your ordinary computer book, not by a long shot. There is a long journey ahead of you, and I will be with you every step of the way.

Buy this book. I am on your side.

Harley Hahn

Harley Hahn's

The Internet Complete Reference, Second Edition

Harley Hahn

Osborne McGraw-Hill
Berkeley New York St. Louis San Francisco
Auckland Bogotá Hamburg London Madrid
Mexico City Milan Montreal New Delhi Panama City
Paris São Paulo Singapore Sydney
Tokyo Toronto

Osborne **McGraw-Hill**
2600 Tenth Street
Berkeley, California 94710
U.S.A.

For information on translations or book distributors outside the U.S.A., or to arrange bulk purchase discounts for sales promotions, premiums, or fundraisers, please contact Osborne **McGraw-Hill** at the above address.

The Internet Complete Reference, Second Edition

Copyright © 1996 by Harley Hahn. All rights reserved. Printed in the United States of America. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but they may not be reproduced for publication.

1234567890 DOC 99876

ISBN 0-07-882138-X

TK5105
.875
I57H34
1996

Executive Editor
Scott Rogers

Editorial Assistant
Daniela Dell'Orco

Project Editor
Janet Walden

Copy Editor
Lunaea Hougland

Proofreader
Pat Mannion

Indexer
Wendy Murdock

Computer Designer
Richard Whitaker

Quality Control Specialist
Joe Scuderi

Illustrator
Marla J. Shelasky

Cover Design
Compass Marketing

Information has been obtained by Osborne **McGraw-Hill** from sources believed to be reliable. However, because of the possibility of human or mechanical error by our sources, Osborne McGraw-Hill, or others, Osborne **McGraw-Hill** does not guarantee the accuracy, adequacy, or completeness of any information and is not responsible for any errors or omissions or the results obtained from use of such information.

96-148419

LIBRARY OF CONGRESS
MAR 28 1996
COPY
COPYRIGHT OFFICE
9

About the Author

Harley Hahn is an internationally recognized writer, analyst and consultant. He is the author of fourteen books, including the best-selling *The Internet Complete Reference* and *The Internet Yellow Pages*, and the highly regarded *Assembler Inside & Out*. His Unix books include *The Unix Companion*, *A Student's Guide to Unix*, *Peter Norton's Guide to Unix* and *Mastering Xenix on the IBM PC AT*.

Hahn has a degree in Mathematics and Computer Science from the University of Waterloo, Canada; and a graduate degree in Computer Science from the University of California at San Diego. Before becoming a professional writer, he studied medicine at the University of Toronto Medical School.

Hahn enjoys writing computer books, because "I get to sleep in, and I like telling people what to do."

Hahn does not live in a converted farmhouse in Connecticut with his wife, three children, and a Labrador Retriever named Rolf. Nor does he commute frequently to New York.

His favorite pajamas are green.

To Wendy

Contents at a Glance

1 Introduction 1
2 Understanding the Internet 9
3 A Tour of the Internet 23
4 Hardware Requirements to Connect to the Internet . . 33
5 Software Requirements and Internet Service Providers . 49
6 Internet Addressing 75
7 Mail 93
8 Using Mail from a Shell Account: **pine** and **pico** . . 125
9 Understanding the Web 163
10 Using the Web 173
11 The Web: Advanced Topics 201
12 Using the Web from a Shell Account: **lynx** 229
13 Introduction to Usenet 249
14 Reading and Posting Usenet Articles 271

15 Using Usenet From a Shell Account: **tin** 297
16 Gopher, Veronica and Jughead 315
17 Using Gopher from a Shell Account: **gopher** 333
18 Anonymous Ftp 349
19 Using Anonymous Ftp from a Shell Account: **ftp** . . 367
20 Archie 387
21 File Types Used on the Internet; Downloading Software . 419
22 Mailing Lists 451
23 Telnet: Logging In to a Remote Computer 469
24 Using Telnet from a Shell Account: **telnet** 477
25 Talk Facilities 489
26 Using Talk from a Shell Account: **talk**, **ytalk** 505
27 IRC (Internet Relay Chat) 521
28 Muds 549
A List of Top-Level Internet Domains 573
Glossary 579
Harley Hahn's Catalog of Internet Resources 603
Index 787

Contents

List of Figures xxiii
Acknowledgments xxvii

1 Introduction 1
What is the Internet? 2
Using the Internet 5
How to Use This Book 5
Do You Need to Know Unix? 7
Should You Learn Unix? 7

2 Understanding the Internet 9
Our Friend the Network 10
The Secret of the Internet: Clients and Servers 12
X Window and X Clients 14
Hosts and Terminals 18
What Is TCP/IP? 20

3 A Tour of the Internet 23
Mail 24

The Web 24
Web Search Engines 25
Usenet 25
Gopher 26
Veronica and Jughead 27
Anonymous ftp 27
Archie 28
Mailing Lists 29
Telnet (Remote Connection) 29
Talk Facilities 30
Internet Relay Chat 30
Muds and Other Imaginary Places 31

4 Hardware Requirements to Connect to the Internet 33
What Do You Need to Use the Internet? 34
Accessing the Internet via Your Local Network 34
Accessing the Internet via the Telephone System 35
Choosing a Computer 37
Choosing a Modem 38
ISDN 41
How ISDN Is Organized 42
What Do You Need in Order to Use ISDN? 44

5 Software Requirements and Internet Service Providers 49
Hosts and Terminals 50
Unix Hosts, VT-100 Terminals, and Telnet 51
Shell Accounts and PPP Accounts 52
The Nature of a PPP Account 54
The Nature of a Shell Account 55
Which Type of Account Is Right for You? 57
The True Story of the Three Brothers 57
PPP and SLIP 58
Internet Service Providers 60
Online Information Services 61
Choosing an Internet Service Provider 63
What Software Do You Need? 66
TCP/IP and PPP Software 68
Installing and Configuring TCP and PPP 69
Where to Get Internet Client Programs 72

6 Internet Addressing 75
Standard Internet Addresses 76
Understanding a Domain Name: Sub-Domains . . . 77
Variations on the Standard Internet Address Format 78
Top-Level Domains 81
Pseudo-Internet Addresses 83
IP Addresses and DNS 84
UUCP 86
UUCP Addresses and Bang Paths 87
Simplified UUCP Addressing 88
Sending Mail to Other Networks 89

7 Mail 93
SMTP: The Basis for the Internet Mail System 94
How You Access the Mail System 95
Which Mail Program Should You Use? 96
An Example of Moving Mail Around the Net 98
Understanding Mail Headers 99
Signatures 103
Mail Addresses 104
Sending Mail 106
Sending Copies of a Message 109
Reading Mail 111
Hints for Managing Your Mail 112
Replying to a Message 113
Forwarding and Bouncing Mail 114
How Mail Is Stored (Folders) 116
The Difference Between Text and Binary Data . . . 117
Using Mime to Mail Binary Data as an Attachment 119
Hints for Practicing Safe Mail 120
Smileys 122

8 Using Mail from a Shell Account: **pine** and **pico** 125
A Quick History of **pine** and **pico** 126
Using **pico** by Itself 128
Orientation to **pine** 128
Starting **pine** to Send a Message 130
Composing a Message: The Screen 131
Composing a Message: General Commands 133

Composing a Message: Moving the Cursor 135
Composing a Message: Editing Commands 136
Composing a Message: Advanced Commands . . . 140
Sending Your Message 142
Editing the Header 143
Starting **pine** to Read Your Mail 145
The Main Menu 147
The Index 150
Selecting a Message to Read 151
Reading a Message 152
Deleting a Message 155
Replying to a Message 155
Forwarding and Bouncing a Message 156
Saving, Printing, and Piping a Message 157
Using the Address Book 158
Customizing **pine** 159

9 Understanding the Web 163
Orientation to the Web 164
How Did the Web Start? 165
How the World Wide Web Turned into "The Web" 166
What the Web Really Is 167
The Web Is the Fourth Attempt 167
Why the Web Grew 169

10 Using the Web 173
What Is It Like to Use the Web? 174
Links, URLs and Hyperspace 177
Words to Avoid 180
Web Pages and Home Pages 181
Telling Your Browser Which URL to Use 183
Reading a Web Page 184
Links, Forms and Image Maps 185
The History List and the Bookmark List 188
Images 189
Sounds and Video 190
Updating Information Automatically 192
Web Directories and Search Engines 193
Customizing Your Working Environment 196
Distributed Applications: Java 198

11 The Web: Advanced Topics 201
URLs and Schemes 202
URLs and Host Names 203
URLs and Port Numbers 204
Pathnames 205
Using Your Browser to Access Anonymous Ftp . . . 208
Using Your Browser to Access a Gopher 209
Using Your Browser to Read Usenet Articles 210
Using Your Browser to Send Mail Messages 211
Using Your Browser to Initiate a Telnet Session . . . 212
Hint for Using URLs 213
Why Hypertext Is Important 216
Hypertext and HTML 217
An Example of HTML 219
The Nature of HTML 222
Making Your Own Home Page 223
Putting Your Home Page Onto the Net 226
Default File and Directory Names 227
Unix File Permissions 228

12 Using the Web from a Shell Account: **lynx** 229
Why **lynx** Is So Cool 230
A Quick History of **lynx** 231
Starting 231
Stopping 232
Displaying Help 233
Generally Useful Keys 233
How **lynx** Displays a Page 234
Working with URLs 235
Navigating 237
Using Your Numeric Keypad to Navigate 239
Special Keys for **vi** and **emacs** Users 241
The History List 242
The Bookmark List 242
Using Forms 243
Saving, Mailing and Printing Files 243
Accessing Anonymous Ftp and Gopher Sites 245
Reading Usenet Newsgroups 246
Sending Mail 247
Customizing **lynx**: Options 247

13 Introduction to Usenet 249
What Is Usenet? 250
Basic Terminology 251
Newsreaders 251
News Clients and Servers 252
What Is the Difference Between Usenet and the Internet? 254
Who Runs Usenet? 255
How Is the News Transported? 256
How Big Is Usenet? 257
The Mainstream and Alternative Hierarchies 258
The Cultural, Organizational and Regional Hierarchies 261
How Many Different Newsgroups Are There? . . . 262
Newsgroup Naming Conventions 262
Moderated Newsgroups 265
What Are the Most Popular Newsgroups? 266
How Are New Newsgroups Started? 267
Frequently Asked Question Lists (FAQs) 270

14 Reading and Posting Usenet Articles . . . 271
The Format of a News Article 272
The Header 272
The Signature 279
Followup Articles 281
Commonly Used Acronyms 284
Usenet Slang 284
Foo, Bar and Foobar 287
Rtfm 287
Smileys 289
How Does Your News Server Keep Track of All the Articles? 291
The Format of Your **newsrc** File 292
Posting Your Own Articles 294
Flames and Good Behavior 294
The Final Word 295

15 Using Usenet From a Shell Account: **tin** 297
A Quick History of **tin** 298
Overview of the **tin** Newsreader 298

Customizing Your Working Environment 300
Starting and Stopping 300
The Newsgroup Selection List 303
Selecting a Newsgroup 304
The Thread Selection List 305
Selecting a Thread to Read 306
Reading Articles 306
Posting an Article 308
Commands That Are Always Available 310
Commands to Use While Selecting a Newsgroup 311
Commands to Use While Selecting a Thread to Read 312
Commands to Use While Reading an Article 313

16 Gopher, Veronica and Jughead 315
What Is the Gopher? 316
A Quick History of the Gopher 317
Which Gopher Client Should You Use? 318
Traveling Through Gopherspace 320
Gopher Addresses 321
History Lists and Bookmark Lists 322
Different Types of Resources 323
Saving, Printing, Mailing and Downloading 325
What Is Veronica? 325
Using Veronica 326
Specifying Words for a Veronica Search 328
Using Jughead to Search Gopherspace 330

17 Using Gopher from a Shell Account: **gopher** 333
Starting the **gopher** Program 334
Stopping, Displaying Help, Shell Commands, Redrawing the Screen 335
Understanding a **gopher** Menu 335
Using Menus 340
Advanced Menu Commands 342
Bookmarks 344
Saving, Mailing and Downloading 345
Summary of **gopher** Commands 346

18 Anonymous Ftp 349
What Is Ftp? 350
Anonymous Ftp 351
The Importance of Anonymous Ftp 352
Using a Browser for Anonymous Ftp 354
Using an Ftp Client for Anonymous Ftp 355
Files 358
Directories 360
Finding Your Way Around a New Computer 362
Requesting Anonymous Ftp Services by Mail 362

19 Using Anonymous Ftp from a Shell Account: **ftp** 367
Starting the **ftp** Program 368
A Second Way to Start the **ftp** Program 372
An Overview of the **ftp** Commands 374
The Basic **ftp** Commands 375
The Connection **ftp** Commands 377
The Directory **ftp** Commands 377
The File Transfer **ftp** Commands 381
Setting **ftp** Options 383
A Typical Anonymous Ftp Session 385

20 Archie 387
What Is Archie? 388
How Does Archie Work? 389
Using an Archie Client 390
Specifying a Search Pattern 392
Using an Archie Client From a Shell Account: **archie** 394
Using an Archie Server Interactively 398
Setting Variables With an Archie Server 399
Setting the **search** Variable 404
Performing a Search 405
Mailing the Results of an Archie Search 405
Commands to Use While Reading Archie Server Output 406
Using the **whatis** Database 407
Displaying Help Information 408
A Summary of Archie Server Commands 410
A Typical Session With Archie 412
Using Archie by Mail 416

21 File Types Used on the Internet; Downloading Software 419
The Five Important Ideas Regarding File Types . . 420
File Types 420
File Extensions for Common File Types 421
Using a Sound, Picture or Video File 421
Using a Document or Postscript File 424
Text Files and Binary Files 425
Compressed Files and Archives 426
Using Mime to Encode Binary Files for Mailing . . 427
Using Uuencode to Encode Binary Files for Usenet 428
Downloading Software for Windows and DOS . . . 430
Important File Types for Windows and DOS Users 434
Background Information About Macintosh Files 434
Downloading Software for a Macintosh 436
Important File Types for Macintosh Users 439
Working With Unix Compressed Files (**Z** and **gz**) 440
Working With Unix Archives (Tarfiles) 442
Compressed Tarfiles 446
Uploading and Downloading Files with a Shell Account 447

22 Mailing Lists 451
What Is a Mailing List? 452
How Are Mailing Lists Different From Usenet Newsgroups? 452
Moderated and Unmoderated Mailing Lists 454
How Mailing Lists Are Administered 454
Bitnet 455
Bitnet Mailing Lists and Usenet 456
Sending Mail to Bitnet 457
Finding Mailing Lists 458
Listserv, Listproc and Majordomo 460
The Two Types of Mailing List Addresses 460
Listserv: Subscribing and Unsubscribing 462
Listproc: Subscribing and Unsubscribing 464
Majordomo: Subscribing and Unsubscribing 466
Manually Administered Mailing Lists 466

23 Telnet: Logging In to a Remote Computer . . . 469
What Is Telnet? . . . 470
Basic Concepts . . . 472
Using Your Telnet Client . . . 472
Connecting to a Specific Port Number . . . 474
More About Port Numbers . . . 475

24 Using Telnet from a Shell Account: **telnet** . . . 477
Starting the **telnet** Program . . . 478
A Second Way to Start **telnet** . . . 480
The **telnet** Escape Character: CTRL-] . . . 482
Summary of Starting and Stopping **telnet** . . . 482
Using **telnet** Commands . . . 483
Job Control . . . 484

25 Talk Facilities . . . 489
Overview of Talk Facilities . . . 490
Chat Rooms . . . 492
Public Events . . . 492
Privacy . . . 492
White Pages Directory . . . 493
A Personal Address Book . . . 493
Embedded HTML . . . 493
Sounds . . . 493
Pictures . . . 493
Video . . . 494
File Transfer . . . 494
Answering Machine . . . 494
Remote Control . . . 494
How Does It All Work? . . . 494
What Do You Need to Use a Talk Facility? . . . 495
Types of Internet Talk Facilities . . . 496
The Unix **talk** Program . . . 496
Internet Relay Chat (IRC) . . . 496
Muds . . . 496
Web-Based Chat Rooms . . . 497
Bulletin Board Systems (BBSs) . . . 497
Talkers . . . 498
Direct Talk Connections . . . 498
Telephone-Like Systems . . . 499
Video Conferencing . . . 499

Web-Based Talk Facilities 500
Guidelines for Text-based Talking 502

26 Using Talk from a Shell Account: **talk**, **ytalk** 505
Introducing the **talk** Program 506
The **talk** Screen 508
Problems with Addresses and What to Do 510
The "Checking for Invitation" Problem and What to Do 512
Refusing to Talk 513
The **ytalk** Program 515

27 IRC (Internet Relay Chat) 521
What Is IRC? 522
IRC Clients 524
IRC Clients for a Shell Account 525
Channels 527
Nicknames 528
Watching a Conversation 529
IRC Commands 531
Basic Commands 533
Help Information 537
Learning More About IRC 538
Using a Nickname 538
Commands for Talking to Specific People 539
Displaying Information About People 542
Using Direct Connections with DCC 543
Bots 545
Summary of Important IRC Commands 546
Harley Hahn's Rules for Safe IRC 546

28 Muds 549
What Is a Mud? 550
Mud Clients 554
Mud Clients for Shell Accounts 555
Social Muds and Adventure Muds 555
LPMuds, DikuMuds and TinyMuds 557
How to Choose a Mud 558
Getting Started on a Mud 560
Talking 565
Illusion and Reality 568
Getting Along with People 569

Moving Around 569
Player Killing 569
Mud Sex 570

A List of Top-Level Internet Domains 573

Glossary 579

Harley Hahn's Catalog of Internet Resources 603

Index 787

List of Figures

2-1: A local area network 11
2-2: A wide area network 12
2-3: An X Window display 17
2-4: A time-sharing system 19
3-1: Summary of internet resources 32
6-1: Organizational top-level domains 81
6-2: Examples of geographical top-level domains 82
7-1: An example of sending mail over the Internet 98
7-2: A typical mail message, showing the full header 100
7-3: Summary of time zone names 101
7-4: Composing a message with Eudora 107
7-5: A summary of incoming mail with Eudora 111
7-6: Replying to a message with Eudora 114
8-1: Composing a message with **pine** 132
8-2: **pine**: Composing a message, general commands 134
8-3: **pine**: Composing a message, moving the cursor 137
8-4: **pine**: Composing a message, editing commands 140
8-5: **pine**: Composing a message, advanced editing commands 142
8-6: **pine**: Commands to send a message 142

8-7: **pine**: Commands to edit the header 145
8-8: The **pine** main menu 147
8-9: A typical **pine** index 150
8-10: **pine**: Commands for using the index 153
8-11: **pine**: Commands for reading a message 154
8-12: **pine**: Commands for processing a message 158
8-13: **pine**: commands for using your address book 159
9-1: A typical gopher menu 170
9-2: A typical web screen 170
10-1: A typical web page 175
10-2: A personal home page 182
10-3: A web page containing a form 186
10-4: A web page containing an image map 187
10-5: Common file formats for images, sound and video 191
11-1: List of schemes used within URLs 203
11-2: A simple HTML file as interpreted by a browser 220
11-3: A hypertext file being edited with Kenn Nesbitt's WebEdit 224
12-1: **lynx**: A typical screen 234
12-2: **lynx**: Navigating keys 238
12-3: **lynx**: A screen with link numbers turned on 240
12-4: **lynx**: An example using forms 244
12-5: **lynx**: The options menu 248
13-1: A graphical newsreader client 252
13-2: The most important Usenet newsgroup hierarchies 259
13-3: The mainstream Usenet newsgroup hierarchies 260
13-4: The alternative Usenet newsgroup hierarchies 261
13-5: Newsgroups devoted to jokes 263
13-6: A selection of newsgroups related to sex 264
13-7: The 25 most popular Usenet newsgroups 268
14-1: A typical Usenet article 273
14-2: Header lines for Usenet articles 274
14-3: A typical Usenet followup article 282
14-4: Common acronyms used on the Net 285
14-5: A selection of smileys 290
14-6: Usenet newsgroups for complaints 295
15-1: **tin**: A typical group selection list 303
15-2: **tin**: A typical thread selection list 305
15-3: **tin**: A typical article 307
15-4: **tin**: Commands that are always available 310
15-5: **tin**: Commands to use while selecting a newsgroup 311
15-6: **tin**: Commands to use while selecting a thread to read 312
15-7: **tin**: Commands to use while reading an article 313
16-1: Using a browser to access a gopher menu 319
16-2: Using a gopher client to access a gopher menu 319

17-1: **gopher**: A typical gopher menu 336
17-2: **gopher**: Menu item symbols 341
17-3: **gopher**: Summary of commands 346-348
18-1: An ftp client program 357
18-2: Summary of important ftpmail commands 364
18-3: Ftpmail servers 366
19-1: A list of all the **ftp** commands 375
19-2: Summary of the most useful **ftp** commands 376
19-3: A sample anonymous ftp session 386
20-1: A graphical archie client 390
20-2: Important symbols used in archie regular expressions 393
20-3: Summary of important options for the Unix **archie** client 396
20-4: Public archie servers 399
20-5: Archie servers: Typical output from a **show** command 400
20-6: Commands to use when viewing output with **less** 406
20-7: Archie servers: Summary of important commands 410-411
20-8: A typical session with archie and anonymous ftp 414-415
21-1: Master list of common file types 422
21-2: Common file extensions, organized by category 423
21-3: The most important file types for Windows and DOS users 434
21-4: The most important file types for Macintosh users 440
21-5: File extensions used with Unix compressed files and archives 448
22-1: Summary of important Listserv commands 464
22-2: Summary of important Listproc commands 465
22-3: Summary of important Majordomo commands 467
23-1: A telnet client program 474
24-1: Using job control with **telnet** 486
25-1: A text-oriented talk program 491
25-2: A talk program with a multiple-person conversation 491
25-3: Live video over the net, using CU-SeeMe 500
25-4: A typical web-based chat room 501
25-5: Common communication abbreviations 503
26-1: A typical **talk** screen 510
26-2: A **ytalk** screen with a two-way conversation 516
26-3: A **ytalk** screen with a five-way conversation 517
27-1: An IRC client (mIRC) 524
27-2: An IRC client from a shell account (**irc**) 525
27-3: A typical conversation with IRC 530
27-4: IRC: Basic commands 546
27-5: IRC: Commands for working with channels 547
27-6: IRC: Commands for sending messages 547
27-7: IRC: Commands for displaying information about people 548
27-8: IRC: DDC commands 548
28-1: A typical message sent from one mud person to another 551

28-2: A mud client program 554
28-3: Selecting attributes for a new character 563
28-4: Talk commands for DikuMuds and LPMuds 565
28-5: Talk commands for TinyMuds 566

Acknowledgments

My favorite part of writing a book comes at the end, when everything else is finished, and all I have to do is write the final section: the acknowledgments. However, it occurs to me that it may not be as interesting for you to read these acknowledgments as it is for me to write them. After all, I know all these people and you don't.

"Why," I can hear you say, "should I care about all the people who helped you? What is it to me that Joe Such-and-such from the Acme Computer Company gave you a Personal Widget System to use with your research?"

Good point. So, to make it more interesting and rewarding for you to read this section, I have decided to hold a contest. Here is how it works.

From now on, wherever you go—to the supermarket, to school, to a business conference, for a workout at the health club—carry this book with you. Whenever you meet a new person, check to see if his or her name is mentioned in this section. If so, have the person sign the page on which the name appears. Once you have collected the signatures of everyone in this section, send your book to me. The first person to send me a full list of all the signatures will win a nice prize. (Of course, the validity of the signatures will have to verified.)

So, now that it is worth your while to read these acknowledgments, let's push on. To start, I would like to thank my chief researcher, Wendy Murdock. Wendy knows

Chapter Ten

Using the Web

In Chapter 9, I introduced the Web and explained why it is so important and useful. In this chapter, I will show you how the Web works, along with the basic techniques you need to have fun and be productive. I will cover everything you need to understand to use the Web, as well as explain what is involved in having your own web site.

What Is It Like to Use the Web?

In Chapter 9, I explained that the Web is based on hypertext, information that contains links to other information or to services. Underneath, the Web is a huge client/server system (see Chapter 2), with many web servers all over the Internet. To access those servers, you use a client program called a browser that fetches information from a server on your behalf, and then displays the information on your screen. Your browser acts as your interface into the Web, doing whatever is necessary to carry out your requests.

INTERNET RESOURCE *Look in the catalog under Internet Resources for* ***Web: Browsers***

If you have never used the Web before, the idea of hypertext and using a browser will seem a bit strange, so let's start with a few examples. First, here is an example from regular life. You are reading an encyclopedia, looking at an article with the title "Trees". At the end of the article, you see a reference that says "For related information, see Plants". This last line is a link from the "Trees" article to the "Plants" article. To follow this link, you must fetch the appropriate volume of the encyclopedia and turn to the "Plants" article.

Now, if you were using a browser to read the "Trees" article on the Web, the reference to the "Plants" article would be highlighted in some way. For instance, the actual word "Plants" might be displayed in a different color and underlined. To follow the link, you would use your mouse to click on this word. Your web client program would then fetch the "Plants" article—no matter where it was on the Internet—and display it for you on your screen.

Here is another, more complex example. You have a friend who has set up his own web site. He gives you the address, and you decide to check it out. To do so, you start your browser and tell it to contact the address of your friend's web site. (Don't worry about the details, I will explain later.)

In order to carry out your request, your browser contacts the web server at the address you specified. This server honors the request by sending information to your browser, which displays it on your screen. You see messages and pictures designed by your friend. Within the text, most of the words are black on a gray background. However, there are certain words that are displayed in red and underlined. These words represent links that you can click on at any time.

This web site is shown in Figure 10-1. Here is the first paragraph you read:

```
I am a librarian at Stanford University.  Last year,
I went on a bicycling trip in Holland with my girlfriend
Yolanda.  While we were there, we visited my friends
Peter and Karin and their bunny Pixie.  My hobbies are
photography and collecting Elvis memorabilia.
```

This example contains several links (which I have underlined). The first is **Stanford University**. If you click on this, you will jump to the web site for Stanford University. The next link is **bicycling trip**. Clicking on this will display a long description, written by your friend, of his trip and all the places he visited. The following link is **Yolanda**. Click on this and you will find more information about your friend's girlfriend. First, you will see a picture of Yolanda, followed by the words "For more information about Yolanda, click **here**." If you do click on that link, you will jump to Yolanda's own web site.

Now that you see how this works, you might be able to guess what type of information the rest of the links point to. If you select **Peter and Karin**, you will see a picture of Peter, Karin, Yolanda and your friend, standing beside their bicycles outside a restaurant in Amsterdam. Clicking on **Pixie** shows you a picture of Pixie

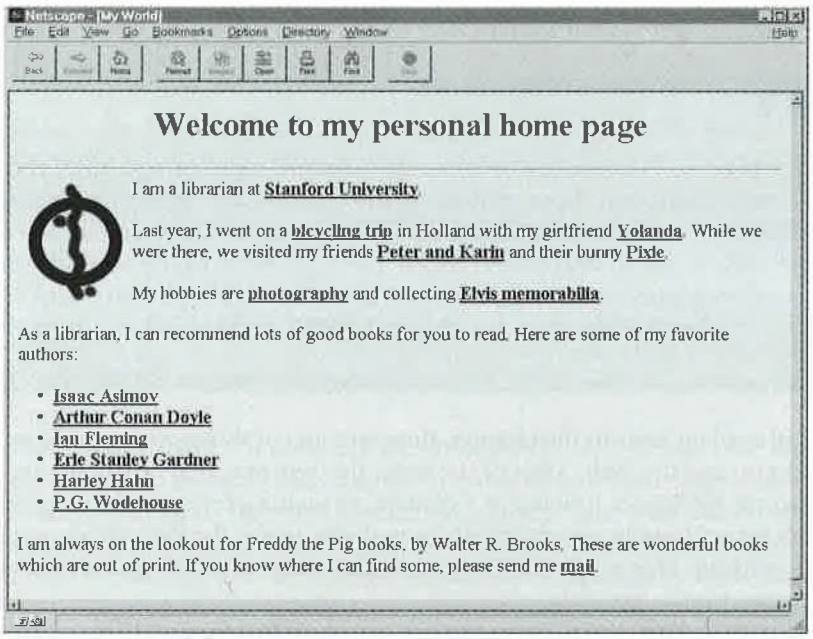

Figure 10-1. *A typical web page*

the Bunny. Clicking on **photography** shows you a small gallery of some of your friend's photos along with a few links pointing to various photographic resources on the Net. Finally, selecting **Elvis memorabilia** takes you to a list of Elvis information sites around the Net, one of which is a guided tour of Graceland (Elvis' home).

Now, the only way to really get a feel for the Web is to use it, but the foregoing example will at least give you an idea of how it works. People create text, and within that text, put links to other text, pictures and various resources. Sometimes links point to something else created by the same person (such as the link to the description of the bicycle trip, or the pictures of Peter, Karin and Pixie). Other links point to resources elsewhere on the Net (such as the links to Yolanda's web site or the Elvis information sites).

This example describes a web site created by one person for his own amusement, and there are many such sites on the Net. However, there are also many web sites created by organizations: companies, schools, government agencies and so on. When you visit such sites, you will find "official" information. For example, the AT&T web site will describe the services and products offered by AT&T. As you might expect, such web sites are better organized, but more formal than the personal ones. For example, the AT&T web site will have slick graphics and all kinds of corporate information. However, what you won't see are descriptions by an AT&T executive of her last trip to Alabama, along with pictures of her family and the pet possum she used to keep as a child.

HINT

The Web is actually a mixture of two very different types of web sites: those created by individuals and those containing information about an organization. This gives the Web a countenance that is unmatched anywhere else in our society. Thus, as you use the Web, you will encounter official, no-nonsense corporate, academic and governmental web sites coexisting peacefully with the highly personal and idiosyncratic creations of people who are free to write and exhibit anything they want.

As I will explain later in the chapter, there are lots of things you can do with your browser as you use the Web. Most of the time, though, you will be doing one of three things: reading some text, looking at a picture, or waiting for something to be copied from a web server to your computer. To be realistic, using the Web means spending a lot of time waiting. This is why, in Chapter 4, I advise you to get yourself as fast an Internet connection as possible.

As you use the Web, the waiting periods are short but frequent. To jump from one web item to another, you use your mouse, click on the next item you want to see, and then wait for it to be loaded. Thus, you will find that you develop a definite rhythm:

click, wait, read... click, wait, read... click, wait, read... The overall timing is affected by what you choose to look at—for example, pictures take much longer to copy than plain text—but just the same, there is a rhythm. If you use the Web during the day when many people are sharing the communication lines, you will wait longer. In the middle of night when there are fewer people, the response times decrease and the pulse of the Web speeds up.

Compare this rhythm (...click, wait, read...) to the rhythm of other popular mental activities. Watching television, for example, is ...click, watch, watch, watch, watch... click, watch, watch, watch, watch. Reading a book is ...turn a page, read, read, read, read... turn a page, read, read, read, read.

HINT

We are shaped by our tools more than we like to admit.

Links, URLs and Hyperspace

Before I discuss the details of using your browser, I would like to explain a few of the important terms and concepts you will encounter as you use the Web.

I have already discussed the idea of hypertext: information that contains links to other information or services. I would like to take a closer look at this idea and answer the question: what exactly is a link?

In the same way that every Internet user has a unique email address, every hypertext item on the Net has an address of its own. This only makes sense. When you tell your browser to fetch and display a particular item, the browser must know where on the Internet that item resides.

The format of such addresses is different from that used for mail addresses. One of the reasons is that the people who first developed the Web wanted it to provide access to all types of information and not just hypertext. Thus, they used an addressing system that is capable of referring to just about any type of Internet information resource. Within this system, we use a special type of address called a UNIFORM RESOURCE LOCATOR, usually referred to as a URL. (The name is pronounced as three separate letters: "U R L".)

Within this system, there is a unique URL for any hypertext item on the Net. Moreover, there are also unique URLs for non-hypertext items from other services, such as gophers, anonymous ftp sites, Usenet newsgroups and wais databases. I discuss these resources in detail in other parts of the book. The important thing to understand here is that, because browsers are designed to access any item that has a URL, you can use your browser to access information from a variety of Internet resources and not just hypertext.

Here are two examples of URLs. (For now, don't worry about the details, I will explain them later in the chapter.)

```
http://www.alan.com/
http://www.bu.edu/Games/games.html
```

Notice that both of these URLs start with **http:**. This means that the URL refers to a hypertext item. (The name **http** stands for HYPERTEXT TRANSFER PROTOCOL, the protocol used to send hypertext data over the Internet.) If a URL refers to a different type of item, you will see a different prefix. For example, a URL that starts with **gopher:** refers to a gopher item, a URL that starts with **ftp:** refers to an ftp resource, and so on. Again, don't worry about the details, I will cover them later.

Each URL refers to one specific Internet resource. To express this idea, we say that the URL POINTS to that resource. For example, the first URL in the above example points to the web site for Alan Colmes, a well-known American radio talk show host. The second URL points to a collection of interactive web games at Boston University.

As you already know, your browser retrieves hypertext data and displays it on your screen. During this process, your browser will recognize the links within the data, and display them so as to make them easy to recognize. For example, your browser may display all the links in a special color, or it may underline each link, or both. (In fact, most browsers give you some choice as to how links are to be displayed.)

Take a look at the following example of hypertext in which the links are underlined.

```
If you want to find out about a real cool guy, check
out Alan Colmes, the famous radio star.  Or, if
you have some extra time, try playing some of
the interactive web games at Boston University.
```

Suppose these two links represent the URLs in our last example. That is, suppose the first link, **Alan Colmes**, corresponds to the URL **http://www.alan.com/**, while the second link, **interactive web games**, corresponds to the following URL: **http://www.bu.edu/Games/games.html**.

If you want to see the information at either of these web sites, all you have to do is select one of the links. (With a mouse, you would click on the actual link.) Your browser will then use the corresponding URL to contact the site and ask for the data to be sent to your computer. When the data arrives, your browser will display it for you on your screen.

HINT

A nice way to think about links is to imagine that underneath each link lies a URL. Thus, each time you select a link, it is as if your browser looks under the link, finds the URL, and does whatever is necessary to fetch the information to which that URL points. With most browsers, you can set an option to have the browser display URLs. Once this option is set, whenever you move your mouse pointer over a link, your browser will show you the corresponding URL. Thus, you can move the pointer around the screen and, each time you pass over a link, your browser will show you the URL that lies "underneath".

When you access a web site by selecting a link, we say that you are FOLLOWING the link. For example, you might follow the **Alan Colmes** link to Alan's web site. Although, technically, it is the browser (a program) that is following the link, we usually talk as if we are performing the action ourselves.

For example, say that you see a guy wearing a T-shirt with a picture of Alan Colmes. You ask, "Where did you get such a radical shirt?" and he says, "I made it from a picture I got from the Web. After I found the picture, I had my browser save it to a file on a floppy disk, which I then took to a place where they printed it on a T-shirt for me." "But how did you ever get the picture in the first place?" you ask. "Oh, that was simple," he responds, "I was looking at a place that had information about talk radio, and I just followed a link to the Alan Colmes web site."

Notice the assumed metaphor. When we use our mouse to select a link, we talk as if *we* are following the link; as if using the Web is like venturing into a foreign land in which we jump from one place to another by following links.

Such observations are more than mere fancy. As human beings we tend to interpret new ideas in light of things we already understand. Thus, it is natural for a human being who is using the Web to imagine himself jumping from one Internet location to another. For this reason, we sometimes talk about NAVIGATING the Web.

The reality of it, though, is that you do not really visit various Internet locations. Each time you follow a link, your browser asks for the information to be sent to *your* computer. Thus, you don't really go anywhere: the various places you "visit" are actually brought to you. Still, if you can suspend belief long enough to think of your screen as a window into the Net, it does seem as if the view changes each time you click on a link.

The word we use to describe all of this is HYPERSPACE. Technically speaking, hyperspace consists of all the possible sites you can access using the Web. Of course, this is a metaphor: hyperspace is not a real place that exists in time and space. Still, it is intriguing to imagine that, in the future, you might connect yourself to the Net in such a way that you actually feel as if you are in a different dimension. And, as you follow the links, you might actually feel as if you are traveling in a strange, alien world. In that sense, hyperspace may one day become a real place in which we may spend all our spare hours. In the meantime, don't forget to stay grounded on Planet Earth and pay your telephone bill.

Words to Avoid

In 1982, William Gibson published a science fiction novel called *Neuromancer* in which he described a futuristic computer-mediated world called "cyberspace". At the time, Gibson was fairly ignorant about computers, so his speculation about computers, humans and the future was both annoying and fascinating.

In the early 1990s, as the Internet and the commercial online services began to grow, people started to refer to these new systems as CYBERSPACE. Later, new expressions for the same idea—the INFORMATION SUPERHIGHWAY, the INFOBAHN, and so on—became popular, especially after they were popularized by the American Senator and Vice-President Al Gore.

Then, in 1992, Jean Armour Polly wrote an article introducing the Internet for a librarians' journal. At the time, Brendon Kehoe had just published a landmark article entitled *Zen and the Art of the Internet* and Polly wanted a title that captured the same tone. She decided to choose something nautical (because the "Net" reminded her of a fishing net). After discarding a number of titles, she came up with one that invoked the idea of "fun, changing conditions, and danger". That title was *Surfing the Internet*.

To Polly (who was a qualified librarian), the word "surfing" captured the essence of using the Internet: "The fun was in trying to find something in the same way that a librarian does research. The danger was that the information might be faulty, old or incorrect. And because conditions on the Internet were always changing, you might have trouble finding your way back to a particular source. Even if you could find your way back, the original information might not be there any longer."

After publishing *Surfing the Internet*, Polly revised it and made it available for free on the Internet. The article quickly became the most popular Internet document of its time. What happened next surprised even Polly. The term SURFING became part of the popular culture, but not as Polly originally intended. Instead, the term came to refer to the activity of browsing the Internet.

Today, Jean Armor Polly is a highly respected Internet expert, sometimes referred to as "Mother Internet". She is, perhaps, the most famous woman on the Internet. (If I weren't so modest, I would tell you who the most famous man is.) However, along with William Gibson and Al Gore, she has the dubious distinction of introducing into the language a word that has become trite and overused.

So here is my advice. I encourage you to tell people that you know how to "navigate the Web" and that you spend time on "the Net". And, if you really want, you can even talk about "hyperspace" (as long as you don't use the word more than once in a single conversation). However, please don't ever let me hear you refer to "cruising the information superhighway", or "surfing the Net" or "hanging out in cyberspace". Such words are to be avoided as they will instantly mark you as someone who has no idea what he is talking about.

HINT

Stay away from meaningless words and expressions that ignorant people use to describe the Net. Remember, as one of my readers, you are a highly intelligent, knowledgeable and motivated individual—not a clueless goober.

Web Pages and Home Pages

Hypertext information is stored in files, and a URL that points to hypertext is actually the address of a specific file on a particular computer. Each time you select a link that points to a hypertext file, your browser will contact the web server at the appropriate site and request a copy of the file specified in the URL.

The contents of a single file of hypertext is called a PAGE or, more formally, a WEB PAGE. Don't get confused, however, by thinking that a web page is like a page in a book. A web page can be any size at all. Some people will design a web site to consist of a single long page. Other people will break the same information into a number of smaller pages (each in its own file), with links between the pages. Thus, a page may be only a few lines, or it may be so large as to contain many screenfuls of information.

Since pages are the logical units of hypertext, you will often hear people referring to an entire web site as a "page". For example, you might overhear a young child ask his mother, "Is it true that Santa Claus has his own web page?" This is a common way to speak, even though many web sites (including Santa Claus') actually have a number of pages.

When your browser requests and waits for a copy of the page from a web server, we say that the browser LOADS the page. (The term comes from the word "download", which means to copy information from another computer to your computer.) Thus, you might hear someone say, "I'll be with you in a moment, just let me get this web page loading."

Another term you will encounter frequently is HOME PAGE. This term has two distinct meanings. First, a home page is the main page for a particular web site. For example, many people have their own web sites which they maintain for fun. These are called home pages. (Figure 10-2 contains an example of such a home page.) With

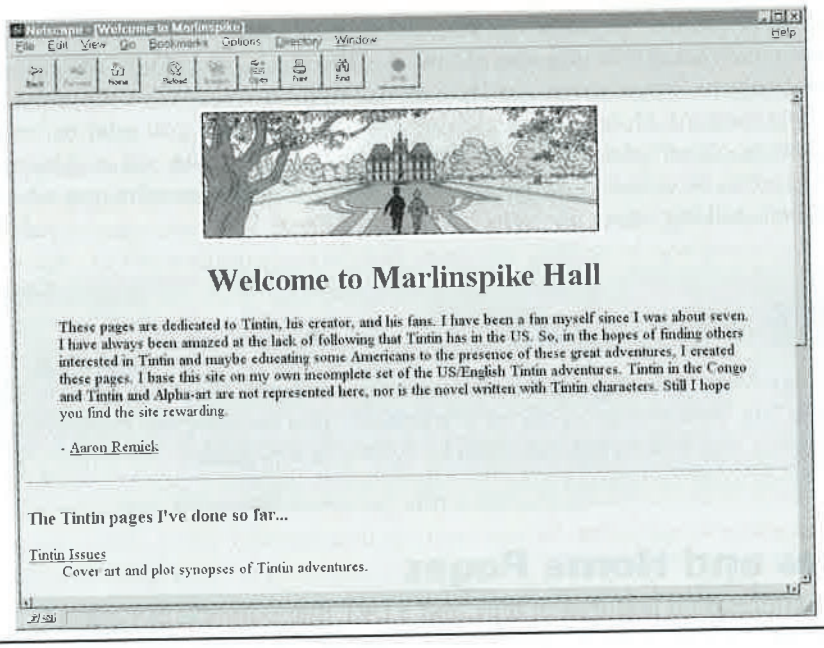

Figure 10-2. *A personal home page*

an organization (such as a company, university or government department), the home page will be less personal and will act as a main menu from which you can navigate to other pages.

Thus, if you were to take a look at IBM's home page, you would expect to find a main menu pointing to various other pages. If you were to look at Norman Sweetcream's home page, you would expect to find a more personal, idiosyncratic offering, perhaps only a single page.

The second meaning of the term "home page" has to do with your browser. Some people like to start each web session by connecting to the same web page. For example, each time you start your browser, you may want to connect to your favorite humor page to see if there are any new jokes. Most browsers allow you to specify the name of a page that should be loaded automatically each time you start the browser. Within your browser, this page is called your home page.

So don't be confused. To your browser, your "home page" is the page you want to be loaded automatically each time you start the browser. To the Internet community, your "home page" (if you have one) is a specific hypertext page, created by you, as your own personal web site. (I will discuss how to create your own home page later in the chapter.)

INTERNET RESOURCE *Look in the catalog under Internet Resources for Personal Home Pages*

Telling Your Browser Which URL to Use

Each browser works a little differently, but there are certain characteristics that all browsers have in common. As you read the next few sections, follow along with your own browser, so you can see how the details work on your particular system. In the examples, I will mention details pertaining to the Netscape browser, but things should work pretty much the same no matter which browser you are using.

(Note: In the following sections I concentrate on graphical browsers, such as the ones you would use with a PPP account running under Windows, Macintosh, OS/2 or X Window. If you have a Unix shell account, you will probably use the Lynx browser, which I describe separately in Chapter 12.)

Some browsers will automatically load a specific web page for you each time you start the program. This is a feature you can turn off or on and, by default, it may be turned on when you install the program. The web page that is loaded automatically is called your home page. (However, as I explained in the previous section, be sure not to confuse this type of "home page" with a personal home page.) If your program has this feature, and you decide to leave it turned on, you can set your home page to any URL you want. For example, if you have a favorite humor site on the Web, you can have your browser connect to it automatically, so you can check to see if there are any new jokes.

The idea of having a home page from which to start is one that seemed to make sense when browsers first became popular. At the time, the gopher system (see Chapter 16) was widely used and, with a gopher, it makes sense to have a home base. The Web, however, is a lot different and you really don't need to start from the same place each time. For this reason, most people find the auto-loading of a home page annoying and choose to turn off this feature. Still, it is there for you if you want it.

In general, it is up to you to tell your browser where you want to go. You can do this in one of three main ways. First, you can type a specific URL. To do this, you can pull down a menu item that allows you to "open" a new URL. With Netscape, for example, you pull down the "File" menu and choose the "Open Location" item. A dialog box will pop up and wait for you to enter a URL. Alternatively, your browser will have a box, probably near the top of the screen, into which you can enter a URL directly. Simply move your mouse pointer to the box and click the mouse. You can now type a new URL and press ENTER.

Another way to choose a URL is to use one of the built-in features of your browser. Most browsers come with pre-installed URLs to help you connect to a few useful resources. For instance, the Netscape browser has several buttons near the top of the screen. These buttons are assigned to specific resources and, to connect to one of these resources, all you have to do is click on a button. For example, one of the buttons says,

"What's Cool!" When you select it, you will be connected to a list of "cool" web sites maintained by the Netscape company.

The third way in which you can tell your browser what URL to follow is to choose an item from your "bookmark list". A bookmark list is a collection of URLs that you yourself have compiled. As you navigate around the Web, you will encounter sites that are especially interesting to you. Your browser will have an easy way for you to save the URLs of such sites in a special list. At any time, you can recall the list and select any item you want. I will talk about your bookmark list in more detail later in the chapter. For now, I just want you to know you can create such a list and that it is an easy way to jump immediately to a particular web site.

Reading a Web Page

Once you specify a URL, your browser will begin to load the web page at that address. You will find that many web pages contain pictures as well as text and, compared to text, pictures can take a long time to load. For this reason, most browsers will load the text first and then work on the pictures. This allows you to start reading as soon as possible.

If you change your mind about wanting the entire page, you can stop the loading whenever you want. Most browsers have some type of "Stop" button which you can click with your mouse. Alternatively, pressing the ESCAPE key will also work with most programs. If you stop loading a page in this way, your browser will display the parts of the page that were already received, so if you see what you want, there is no need to wait for the entire page to be loaded.

HINT

With most browsers, you do not have to wait until the entire page is loaded to start to select a new link. Even though the current page may still be loading, you can move your mouse pointer to a link and click on it. The browser will stop loading the current page and jump to the new one.

Once a web page is loaded, you can read it at your own speed. If it is too large to fit on your screen, you can move around by using your mouse (click on the scroll bars) or your keyboard (with the PAGEUP and PAGEDOWN keys).

Aside from reading, there are several other things you can do with a page once it is loaded. You can save it to a disk file, print it or mail a copy to someone. You can also save the URL for the page in a special collection of URLs called your "bookmark list" (explained later in the chapter). Saving a URL on your bookmark list allows you to find it again whenever you want. The details of performing these operations vary from one browser to the next. Usually, though, all you will need to do is select an item

from a pull-down menu, and specify the appropriate information (file name, mail address and so on).

Links, Forms and Image Maps

Once you have loaded a web page, you can navigate by selecting one link after another. Just move your mouse pointer to a link, click on it, and wait.

HINT

There is an easy way to tell if a word or small picture is a link. Turn on the browser option that tells it to display URLs. Now whenever you move your mouse pointer over a link, your browser will display the corresponding URL. If you are not sure whether or not something is a link, all you have to do is point to it and see if a URL appears.

Most links are simple. They will either be highlighted words or a small graphical element such as a button or a picture. There are, however, two other types of links which are more complex that I want to discuss.

The first type of special link is a FORM. A form is a facility that allows you to enter information to be sent back to the remote web site for processing. For example, say you are visiting the home page of someone who has set up a "guest book". The guest book is set up so anyone who visits the page can leave a personal message, and subsequent visitors can read what other people have written.

When you look at the home page, you will see a box into which you can type a comment. This is the form. Beside the box will be a word or a small picture you can click to send the comment once it is typed. For example, you may see a button that says "Submit". All you have to do is type your comment and then click on the "Submit" button. Figure 10-3 shows a web page that has such a form.

Originally, the idea of a form was developed so that users could submit information which would be sent to the remote web site to be processed, just like a regular form on paper. Although this is still one of the biggest uses for forms, people have found more imaginative ways to use this facility. For example, there are interactive web games in which you can make a move or specify a strategy by filling out a "form" and submitting it.

You might wonder, what happens to the information when it reaches the other end? How is it processed? The answer is, the web server that accepts the information sends it to a special program called a CGI SCRIPT. (The name stands for COMMON GATEWAY INTERFACE.) Thus, whatever you type into the form is sent to a CGI script at the other end that receives the information and processes it in an appropriate manner. This means the person who designed the form must also have set up a CGI script to handle the incoming data. As a user, you do not need to worry about the

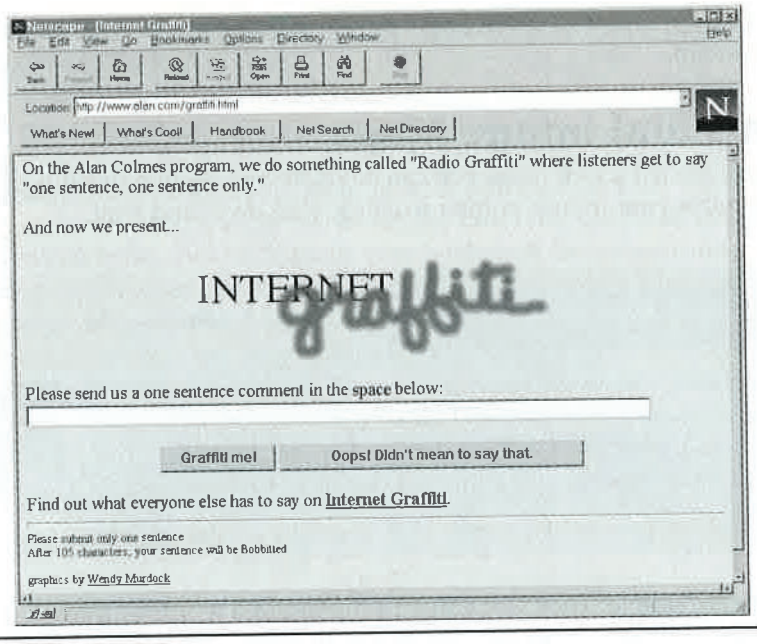

Figure 10-3. *A web page containing a form*

details. However, if you are developing a home page of your own and you want to use a form, you will have to supply a CGI script to handle the incoming data.

INTERNET RESOURCE *Look in the catalog under Internet Resources for* ***Web: CGI (Common Gateway Interface)***

The second type of special link I want to mention is an IMAGE MAP. This is a picture of some type (typically a photo or a drawing) in which the various parts of the image act as separate links. For example, there is a weather report service that will send you a page containing a map of the United States. To request a weather report for any area of the country, all you have to do is click on that part of the map. (Figure 10-4 shows an example.) One common design is to have an image map consisting of several small pictures, each of which is actually a link to a different web page.

How does an image map work? As you move your mouse pointer around the image, your browser keeps track of the coordinates (similar to longitude and latitude on a map). When you click on a particular location, your browser will send the

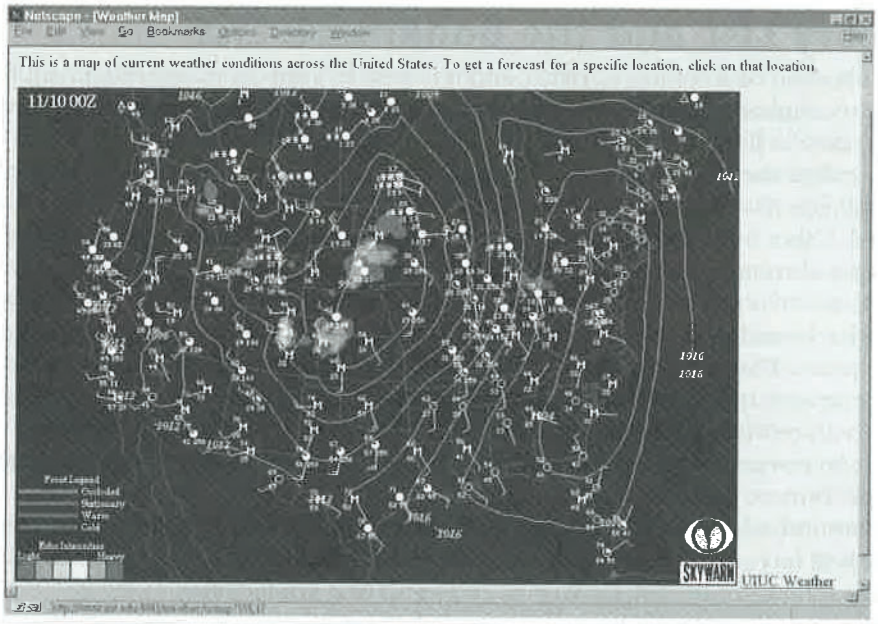

Figure 10-4. *A web page containing an image map*

coordinates of that location to the remote web server. The web server will then look at the value of the coordinates and send back the appropriate information.

HINT

One problem with image maps is there is no sure way of knowing exactly where the links are. Some browsers will show you the actual coordinates as you move your mouse pointer around the image, but there may be no other clues. Indeed, it can sometimes be hard to recognize a particular picture as being an image map.

For this reason, a good web page designer will put in some text to show you are looking at an image map. For example, you may see something like, "To visit one of our electronic stores, just click on the part of the picture that shows what you want to buy."

INTERNET RESOURCE *Look in the catalog under Internet Resources for* ***Web: Image Map Tutorial***

The History List and the Bookmark List

Hyperspace can be a confusing place, and it is easy to jump from one link to another without remembering where you have been. Still, there will be times when you will want to revisit a link you left behind. To help you, your browser keeps a record—called the HISTORY LIST—of the links you have visited. Some browsers keep every link you have visited in the current session, regardless of how much you have branched. Other browsers keep only those links that trace your path from the first page to the current page, with no branching.

There are two ways to use the history list. First, most browsers have an easy way to move backward or forward one link at a time. Typically, there will be two buttons you can press. They will be marked "Back" and "Forward", or have left and right pointing arrows, or something similar. When you click on the "Back" button, your browser will return to the last page you loaded. If you want to move back even further, you can press the button as many times as you want. Once you have pressed the "Back" button, you can press "Forward" to display a subsequent page.

The second way to use the history list is to choose an item from it directly. Your browser will have a way to display the whole history list on your screen. You can then select any item from the list, click on it, and jump to it immediately.

The history list is useful for moving back and forth within a single web session, but there will be many times when you will want to save a web address permanently. For this reason, all browsers have a facility that will save the URL of the current page to a BOOKMARK LIST. Whenever you want, you can display the bookmark list, select an item, and jump to it directly. Unlike a history list, your bookmark list only contains URLs put there by you. Moreover, a bookmark list is permanent. Your browser will automatically save it to a file, so as to be available from one session to the next.

Most browsers have a way for you to edit your bookmark list. At the very least, you should be able to modify the description of any item on the list, as well as delete any unwanted items. Some browsers even allow you to organize the items into categories and subcategories, in the same way that you might organize files into folders or directories.

HINT

As your bookmark list grows, it can become unwieldy and filled with unwanted items. I suggest that you take a few moments every now and then to organize your treasures. You never know what the future holds. If you become famous, a major university may want your bookmark list (along with your papers) after you die. An overgrown, disorganized list will reflect badly on you, serving only to create everlasting embarrassment for your family and friends.

Images

One of the nice things about the Web is there are so many visual images. Broadly speaking, there are two types of pictures: inline images and external images. An INLINE IMAGE is a picture that is a part of a web page. For example, you will often see small pictures or icons loaded along with the text of a page. These are inline images. An EXTERNAL IMAGE is a picture that is a web page in its own right and, when loaded, is displayed in its own window.

Here is an example. Imagine an Internet art gallery which is set up to show you pictures of famous paintings. However, a single picture contains so much data that it would take a long time to load. If the gallery were set up like a real gallery—where you walk in and see all the pictures at once—it might take minutes and minutes to load the gallery home page. The solution is to design the gallery so you can preview the pictures by looking at small images (which load quickly). Then, when you decide which picture you want to see, you can tell your browser to load the full-sized version.

In order to implement this system, the web page designer will create miniature versions of each picture in the gallery. These small pictures will be included on the home page as inline images. Thus, whenever you load the home page, you will see a catalog of what is available in the gallery. Each inline image acts as a link to the file that contains the full-sized picture. So to display a full-sized picture, all you need to do is click on the corresponding small, inline image. When you do, your browser will load the picture and display it in its own window. If you want to look at another picture, you can go back to the home page and click on another inline image.

HINT

If you ever want to use the Web, and you don't care about looking at pictures, your browser will have an option to turn off the automatic loading of images. Setting this option is handy when you want to find something fast, because you don't have to wait for the pictures each time you load a new page.

If you have a shell account (see Chapter 5), another alternative is to use Lynx, a text-only browser. (See Chapter 12 for details on using Lynx.) Personally, when I am doing research and I don't care about images, I often use Lynx because it is much faster than a graphical browser.

Images are stored in various formats, the two most common being GIF and JPEG. For convenience, GIF files are given names that end in **`.gif`**. Similarly, JPEG files are given names that end in **`.jpg`**. For example, you might encounter image files named **`harley.gif`** or **`wendy.jpg`**. The GIF format was developed by Compuserve. The name GIF stands for "Graphics Interchange Format". JPEG is named after the organization that developed the format, the Joint Photographic Experts Group.

In order to display an image, your browser must be able to understand the format of the data. For example, to display the file **harley.gif**, your browser would have to be able to understand what to do with a GIF file. In web language, a program that can display a particular type of file format is called a VIEWER. Thus, in order to display GIF files, your browser must have access to a GIF viewer. To display JPEG files, your browser needs a JPEG viewer.

It is likely that your browser will have built-in viewers for GIF and JPEG. If it doesn't—or if you want to be able to display other image formats—you can use an external viewer. Simply find an appropriate program (they are readily available on the Net), install it on your computer, and then tell your browser which file formats the new viewer will handle. To do this, use the menu item in your browser that lets you change the configuration or preferences information. This tells your browser which viewer to use when it encounters a particular file format.

Sounds and Video

In addition to images, you will also find many sound and video resources on the Web. Although there are a number of different file formats, only a few are in common use.

The sound formats you are most likely to encounter are AU and WAV. AU was developed by Sun Microsystems; the name stands for "audio file". WAV was developed by Microsoft for Windows; the name stands for "waveform data". There are also other formats (which I will discuss in a minute) that are used for "real-time audio".

For video, there are three common file formats. MPEG was developed by the Motion Picture Experts Group, Quicktime was developed by Apple, and AVI was developed by Microsoft for Windows. The name AVI stands for "Audio/Visual Interleaved data". ("Interleaved" is a technical term indicating that the data is processed in alternating sequences. Don't worry about it.)

As with GIF and JPEG, each of these types of data is stored in files with special names. AU files are given names that end with **.au**; WAV file names end with **.wav**; MPEG file names end with **.mpg**; Quicktime file names end with **.qt** or **.mov** (movie); and AVI file names end with **.avi**. This means that you can tell what format a file uses just by looking at its name. For reference, the most common file formats for images, sound and video are summarized in Figure 10-5.

To listen to audio files or to watch video files, you need programs that can work with the appropriate data formats. With GIF and JPEG files, you use a viewer to display the file. Similarly, we call programs that interpret sound and video files "viewers" (even though you don't really view sound). You may see other names—for example, Netscape calls such programs "Helper Applications"—but the name "viewer" is a generally-accepted generic term.

Although some of the formats in Figure 10-5 were developed by one company or another, they are all in common use, and viewers are available for all types of computers. For example, although the AU format was developed by Sun, you can

Format	Typical File Name	Origin of Name
GIF	`filename.gif`	Graphics Interchange Format (Compuserve)
JPEG	`filename.jpg`	Joint Photographic Experts Group
AU	`filename.au`	Audio file (Sun)
WAV	`filename.wav`	Waveform data (Microsoft)
Quicktime	`filename.qt`	Quicktime (Apple)
Quicktime	`filename.mov`	Quicktime [Movie] (Apple)
AVI	`filename.avi`	Audio/Visual Interleaved data (Microsoft)
MPEG	`filename.mpg`	Motion Picture Experts Group

Figure 10-5. *Common file formats for images, sound and video*

listen to AU files on a PC or Macintosh, as long as you have a program that can interpret that file format.

With respect to sounds, there are two types: regular sounds and REAL-TIME SOUNDS. When you listen to a regular sound, the entire file is downloaded to your computer before the sound is played. With a real-time sound, the sound is played as it is being downloaded.

The advantage of a real-time sound is you do not have to wait for the entire downloading process to complete before you can start listening. However, the sound itself is not saved. It is played as soon as it is received, and the data is not retained.

Regular sounds, on the other hand, are stored in a file. Thus, you can keep a copy of a regular sound and replay it whenever you want. Moreover, regular sounds tend to be of higher quality, because in order to make a real-time sound load fast enough, some of the data is discarded.

Real-time sounds are good for transmitting large amounts of data that no one really wants to store permanently. For example, you might listen to a broadcast of a sporting event or a radio program via real-time sound. Regular sounds are used for storing more permanent data. For example, you will find many regular sounds available in web sites and anonymous ftp sites around the Internet.

If your browser does not come with a built-in viewer for any of these formats, external viewers are readily available on the Net. When you install a viewer on your computer, all you need to do is tell your browser where the program can be found, so it knows where to ask for help when it encounters that particular file format. (You set up such information using the configuration or preferences menu.)

Regular sounds use the file formats I mentioned above. Real-time sounds use a variety of different formats. The important thing is to make sure you have the correct viewer—that is, playback program—for the type of sound to which you want to listen.

HINT

Sound and video data takes up a lot of room, even more so than pictures. For this reason, such files tend to be large. Indeed, it is not uncommon to spend minutes and minutes loading a (regular) sound or a video clip, only to have it play back in just a few seconds.

In this way, listening to sounds or watching video on the Web is a lot like Life. People complain that it takes a long time to get what they want and that the quality is poor, and yet the whole thing is over much too soon.

Updating Information Automatically

Once you tell your browser to load a web page, you must wait for the information to be sent from the web server to your computer. As the information arrives, your browser (the web client) displays it for you. However, once the page is displayed, it does not change.

Nevertheless, there are times when it is desirable to display information which can be updated automatically. There are two ways in which this can be done.

First it is possible for a web page to be set up so that it tells your browser to reload itself (or a different web page) automatically. For example, a web page might tell the browser to reload the page every 30 seconds. This facility is called CLIENT PULL, because the client program (your browser) is told to "pull" information from the server without your intervention.

An alternate system allows the server to send new data on its own, without your browser having to make a request. This facility is called SERVER PUSH.

With client pull, the connection between the server and your computer is dropped as soon as the web page is finished loading (which is what happens normally). When your browser is ready to ask for more data, it reconnects to the server and issues the request.

With server push, the connection between the server and your computer is kept open indefinitely. This allows the server to send more data whenever it wants.

Client pull and server push are useful in situations where you are looking at data that changes. For example, a financial service could use server pull to update a stock market report every five minutes. An example of client pull is found with certain web-based talk facilities. (A talk facility lets you use the Web to send messages back

and forth to other people.) By using client pull, the talk system can update your window at regular intervals in order to show you any new messages that may have arrived. (For more details about talk facilities, see Chapter 25.)

Client pull and server push will not work with any web page or web server. A web page must specifically request client pull, and a server must be designed to offer server push. In addition, your browser must be aware of these features and must know how to support them. (If your browser does not support these services, the web page will still load, but it will not be updated automatically.)

Web Directories and Search Engines

There are so many items available on the Web, I guarantee you will never run out things to do—as long as you know how to find what you want. However, as I explain in Chapter 1, there is no central authority in charge of the Internet, which means there is no central directory of web sites. Anyone can create (or remove) a web site at any time.

A central directory would be wonderful, as it would allow you to search for anything you want. For example, you might want to look for a certain type of gardening information. Or you might want to get a copy of a free program which is available somewhere on the Net. How do you know where to look?

A good number of people have attempted to solve this problem by compiling a Web directory, and designing a tool to allow users to search the directory for a particular resource. The reason I say "attempted to solve the problem" is that the Web is so huge and so volatile, it is impossible to have a fully comprehensive, up-to-date directory.

Still, web directories do make excellent resources, and if you know how to use them, you can almost always find what you want. This is because web page designers tend to put in links to similar resources. Thus, even if you find only a few resources related to a particular topic, these will often lead you to all the information you want. For example, if you are looking for some specific gardening information, starting with just a few gardening pages will likely lead you to all the links you need.

As you can imagine, it would be a difficult and time-consuming job to compile a comprehensive web directory by hand and to keep it up to date. For this reason, the people who maintain such directories use special programs to do most of the work. These programs search the Web relentlessly, looking for new and updated items. (It is common to see such programs referred to by exotic names, such as "spiders", "worms", "crawlers", "robots" and so on. However, don't worry about the names; they are not important.)

There are a number of free, well-maintained directories available on the Web and, in general, there are two ways to use them. First, some directories have home pages organized into categories. To use such a directory, all you need to do is load the home page and choose the category you want. Within that category, there may be subcategories. Thus, you can select categories and subcategories, until you narrow down your search to find exactly what you want.

Other directories allow you to look for something by using a SEARCH ENGINE. To use a search engine, you use your browser to fill in a form, in which you describe what you are looking for. Once you submit the form, it is sent to the search engine, which quickly looks through the directory, finds what you want (if it exists), and sends back the results.

Regardless of how you access a directory—either by selecting categories or by submitting a request to a search engine—the results are similar: You will end up with a number of links. All you need to do is follow the links to see if, indeed, they contain the information you want.

To help you decide which links are the most useful, the search engine will show you some descriptive information. For example, you will usually see not only a set of links pointing to web pages, but the titles of those web pages as well. Some search engines may also show you part of the actual page. Still others will do their best to rank the results from most to least relevant, based on the search criteria you specified.

Using a search engine for a simple search is easy. Usually, all you have to do is type one or more words into the form, click on the "Submit" button, and wait for the results. However, if you specify a more complex search, you can often get better results.

Most search engines allow you to use various types of search criteria. For example, say you want to find all the web sites that contain a reference to "Harley Hahn". The simplest way would be to search for all the web pages in the directory which contain the two words **Harley Hahn**. However, if this doesn't find what you want, you might want to expand your search by looking for pages that contain either **Harley** or **Hahn**. Or you might want to search for the word **Harley**, followed immediately by the word **Hahn**. Or, perhaps, either the word **Harley** followed by **Hahn**, or the word **Hahn** all by itself. Various search engines allow you to formulate a search in different ways. However, there will be help information, and learning the details for a particular search engine is usually easy.

One set of rules you will often see are those allowing you to combine words using the "and" and "or" operations (like the examples in the previous paragraph). Such searches are often called BOOLEAN SEARCHES.

What's in a Name?

Boolean Search

A boolean search is one that combines search items using "and" and "or" operations. The name originates with George Boole, an English mathematician (1815-1864) who flourished in the mid-nineteenth century. Boole, along with another mathematician, Augustus de Morgan, was one of the chief developers of symbolic logic.

Now, symbolic logic (keep following because I am leading you somewhere) applies mathematical symbols to logic. Using such symbols removes the possibility of verbal ambiguities and simplifies the manipulation of logical relationships. In the history of mathematics, this is considered a big deal. For example, not long after the turn of the century, Bertrand Russell and Alfred Whitehead used and extended Boole's symbolic logic to create the first logical foundation of mathematics. Although this work has since been superseded by more modern analysis, it is considered to be one of the stunning intellectual achievements of the twentieth century.

This same use of logic and symbols was later adapted by computer scientists to create techniques used to design and analyze computing machines. In honor of George Boole, this field of study was called boolean algebra. One important facet of boolean algebra is being able to analyze expressions that involve the "and" and "or" operations. Thus, it has become the custom to use the word "boolean" to describe anything that involves such operations. Hence, it is common for searches defined by "and"s and "or"s to be called boolean searches.

Although many people talk about such searches, few people know what the name means, and even fewer know its origin. (But now you do.)

Aside from simple boolean searches, most search engines allow you to use other, more complex types of search patterns. I suggest you become familiar with three or four different search engines, and take the time to read the help information. With a little practice, you will be able to find just about anything you want.

HINT

Take some time to learn how to use several search engines well. The Web is vast beyond the understanding of any single human being, and being able to search with skill and confidence will open doors which are closed to most everyone else.

INTERNET RESOURCE *Look in the catalog under Internet Resources for* ***Web: Search Engines***

Customizing Your Working Environment

In general, all browsers have the same purpose in life: to act as a client program—that is, your interface—as you use the Web and other Internet services. There are a great many different browsers, and each one has its own characteristics. Nevertheless, all browsers allow you to customize your working environment to some extent.

In order to use your browser well, you should take some time to explore the program itself and its options; to see what you can change and what choices you have. In most cases, you will access the customization features by using a pull-down menu. Look for a menu or submenu named "Preferences", "Options", "Settings" or something similar.

My advice is to take some time and find out how your particular browser lets you modify your working environment. In general, it is safe to change any of the settings, except those which control your actual network connection.

HINT

As you learn how to use your browser, don't be afraid to experiment. What's the worst that could happen? If you really screw things up, you might need someone else to fix the settings for you, or perhaps even reinstall the software. So what? After all, if Humphrey Bogart had been afraid to experiment with his browser, where would he be today?

To get you started, I will discuss the most common ways to customize your working environment. Your browser will have other settings, but the ones I describe here are generic.

- TEXT AND COLORS: You can specify how text should be displayed (for example, size and typestyle), as long as your system allows such controls. You can also set the background and foreground colors used by your browser.
- LINKS: You can specify how links are to be displayed on your screen. You can change their color, underline them, and so on.
- HOME PAGE: Each time your browser starts, it can automatically load a particular page, called a home page. You can turn this feature off and on, as well as specify a URL for the page to be loaded. (Note: Don't confuse the use of the term "home page" in this context with a personal home page. See the discussion earlier in the chapter.)
- WINDOWS: You can change the size and shape of the various windows used to display information. To experiment, load a page with both text and images, and see what happens to each as you change the size of the window. In general, text will wrap automatically to fit in a newly sized window, while images will not change.
- MANIPULATING WINDOWS: Learn how your particular program and GUI (graphical user interface) let you open, close, iconize (shrink) and move windows. In addition, some browsers will let you open a new window whenever you want, so you can load more than one web page at the same time, each in its own window.
- IMAGE LOADING: You can control whether or not images are loaded automatically. If you turn off this feature, things will move a lot faster, as images can take a long time to load. However, you won't see the pictures unless you ask for them specifically.
- VIEWERS: Your browser already knows how to display certain types of data. However, when you encounter a different data type (image, sound, video), your browser will have to call upon another program, called a viewer, to interpret the data. You can find various types of viewers on the Net and install them on your computer. After you do, you must tell your browser about the new viewer and what data types it handles. (Note: Sometimes viewers are referred to as HELPER APPLICATIONS.)

INTERNET RESOURCE *Look in the catalog under Internet Resources for* ***Web: Helper Applications***

- THE TOOLBAR: Some browsers have toolbars (collections of small icons) which allow you to access frequently used operations with a single mouse click. To perform an operation, you can click on one of the small icons, rather than selecting an item from a pull-down menu. If your browser uses toolbars, you will be able to control which toolbars or icons are displayed. Some people like to show all the toolbars all the time. Other people never use them at all.

- BOOKMARK LIST: Your bookmark list is a collection of your favorite links, which you can recall and use whenever you want. Your browser allows you to customize your bookmark list by adding, deleting and editing links. Some browsers even let you organize your bookmark links into folders and subfolders. (For more information about using your bookmark list, see the discussion earlier in this chapter.)
- THE CACHE: Whenever your browser loads a new page, it keeps a copy in a storage area called the CACHE. Using the cache makes it quick to return to a page you have already visited, as your browser can display the data immediately without having to reload it from a distant web server. The cache can be either in memory (which means it disappears when you quit the program) or stored on disk (which means the data is maintained from one work session to the next). If your browser can use both types of cache, you should choose which one you want to use. (A memory cache is faster; a disk cache is more permanent.) You can also control how long items are kept in the cache before they are deleted automatically. (When your browser deletes expired items, we say that it FLUSHES the cache.)

HINT

If your browser uses a disk cache, you may end up accumulating a lot of unused cache files. (Each web page and each external image is kept in a separate file.) For example, as I was writing this chapter, I checked my disk looking for old cache files. One of the browsers I use had created 534 files taking up 4,854,820 bytes of disk space. Another of my browsers had stored 266 files using 1,125,128 bytes. Needless to say, it was the work of a moment for me to delete these superfluous files and reclaim the disk space.

Some browsers allow you to select a menu item to flush the cache immediately. If you are using a disk cache, you should probably do this once in a while, just to get rid of old abandoned files. Alternatively, you can look in the disk directory in which your browser keeps the cache files, and delete all the old files manually. Of course, you should not do this unless you are comfortable with files and directories and know what you are doing.

Distributed Applications: Java

A web page contains information in the form of hypertext, and a web browser displays that information. Once the page is displayed, it doesn't really do anything, it just sits there. However, using special technology, it is possible for web page designers to construct programs that can be sent over the Web along with the regular information. When your browser receives such a program, it can run it while it

displays the hypertext. Being able to distribute web programs in this way adds a lot of functionality to the basic system.

For example, a web program might display graphics, or show animation, or perform calculations. In fact, just about anything that can be done with a program can be attached to a regular web page and distributed over the Net. The important thing about these programs is the flexibility they bring. The programs are not installed on your computer permanently. They are loaded and executed dynamically, whenever you select the web page to which they are linked.

Such programs are sometimes referred to as DISTRIBUTED APPLICATIONS ("application" being a synonym for "program"). Distributed applications are written in a computer language that is specifically designed for creating distributed programs. The language design must ensure that it is impossible to write a program that can cause damage on someone's computer. After all, when a program is sent over the Web and executed automatically, you don't have the same level of control as when you download a program and install it yourself.

To distinguish such programs from regular software, we sometimes refer to them as APPLETS. One of the important computer languages for writing web-based applets is named Java. If you are a programmer, you can think of Java as being a lot like C++ without the pointers.

To use a Java applet, all you need is a browser that is Java compatible. If you have such a browser, you do not have to do anything special. When you select a web page that contains a Java applet, your browser will load and execute the applet automatically along with the rest of the page. If you try to load the same web page with a browser that is not Java compatible, it will display the information in the regular manner and ignore the applet.

Chapter Eleven

The Web: Advanced Topics

Chapter Twenty-Five

Talk Facilities

Imagine a group of children in a classroom, working quietly under the supervision of a gimlet-eyed schoolteacher. What do they do when the teacher turns her back? They whisper furtively to one another and pass notes, hoping the teacher won't see.

Skip ahead twenty years and the same kids are now sitting in a conference room, listening to the Vice President of Marketing go on and on about the sales projections for the winter quarter. Look closely, and you will see one person quietly slip a piece of paper to his neighbor. She opens the paper to read: "He sure is boring, isn't he? He reminds me of my Uncle Arthur."

Like the child of twenty years ago, the man in the meeting is perfectly willing to do whatever is necessary to make personal contact with another person. When he shares his feelings and ideas he feels better. Moreover, he may be interested in establishing a more personal relationship with the woman sitting next to him.

Put human beings together anywhere and, in a short time, they will have figured out a way to talk to one another. So, as you might expect, as an Internet user, there are many ways for you to talk to someone else.

In this chapter, I discuss the ideas involved in talking over the Internet and describe some of the more common services. Two of the services are so important that I have given them chapters of their own: Internet Relay Chat (Chapter 27) and Muds (Chapter 28). In addition, in Chapter 26, I explain how to use **`talk`**, a Unix program for shell account users. However, before you read any of these chapters, do start with this one, so I can make sure you understand the basic concepts.

Overview of Talk Facilities

When we use the word TALK on the Net, we mean any type of communication in which you can send messages back and forth with someone in real time (that is, while you wait). The "talking" may involve text messages or actual voice transmission. For example, you might tell someone, "Last night I was talking to a friend in Australia," even if you and your friend were typing back and forth with no voice. On the Net, anytime you make contact with another person, you are talking.

(You will sometimes see people use the word CHAT. Usually this refers to talking to more than one person at a time, although there is no strict definition.)

The Internet has many different talk facilities available for free. The most basic service lets you type messages back and forth with another person. Whatever you type shows up on the other person's display, and whatever he or she types shows up on your display.

A typical setup will use a window divided into two areas. Your typing is displayed in one part; the other person's typing is displayed in the other part. As you talk, you can see both sides of the conversation at once. Since there are two display areas (one for each of you), you can both type at the same time without getting mixed up. An example of this is shown in Figure 25-1.

Many such systems allow you to talk to more than one person at the same time. When you do, your program will divide the window into as many talk areas as necessary. You can see this in Figure 25-2 which shows Bill Gates talking to four people at the same time.

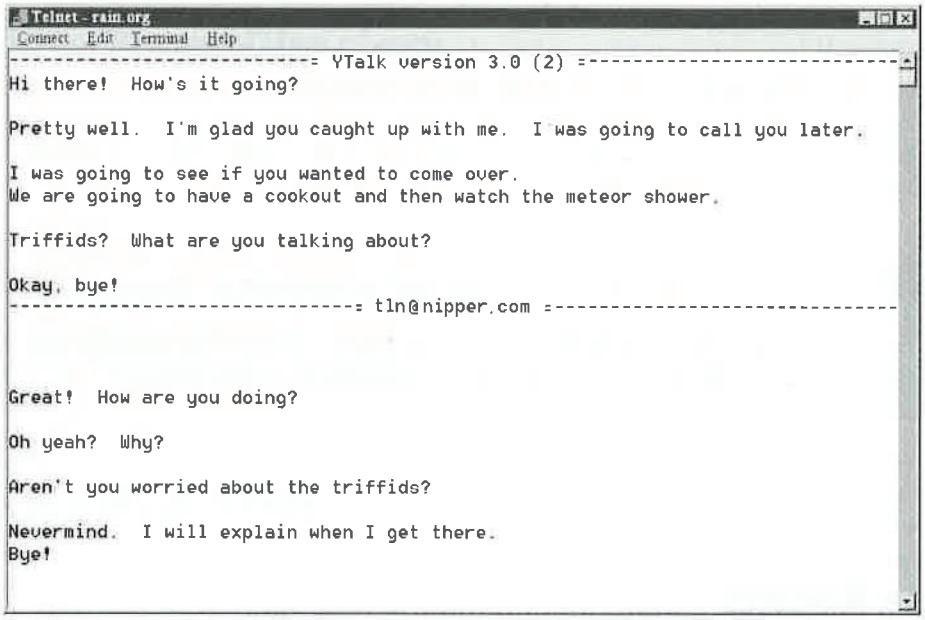

Figure 25-1. *A text-oriented talk program*

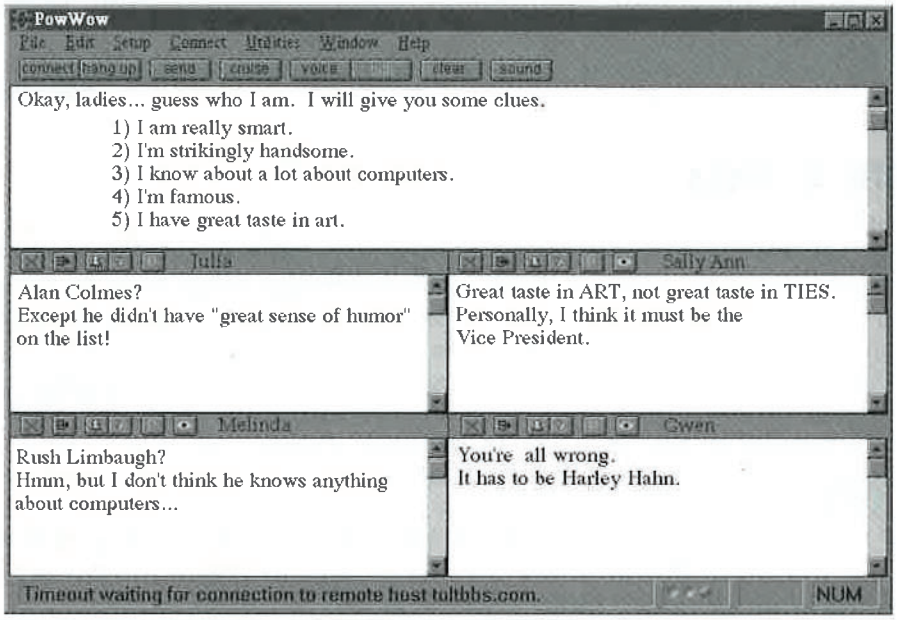

Figure 25-2. *A talk program with a multiple-person conversation*

A more complex talk facility will let you send and receive actual voice sounds, like a telephone. When you use a voice-oriented program over the Net, you can talk with people all over the world for free without paying a long distance charge. However, the quality may not be as good as with a regular telephone. Still, as a beautiful woman in California recently remarked to me, "I'm not going to pay a huge telephone bill just to talk to some stud in Germany for an hour."

HINT

To use a voice-oriented talk facility, your computer must have the appropriate hardware. It must be able to play sounds, and it must have a microphone.

These services—text and voice—are the two most important ones you will find. Some talk facilities have one or the other, some have both. In addition, there are a number of other features you may find.

Chat Rooms

A CHAT ROOM is a facility that allows more than one person to talk at the same time. Chat rooms simulate an environment in which people gather together to talk. Although chat rooms are imaginary, it is common to talk about them as if they really exist. When someone joins the conversation, we might say he or she "enters" the room. Similarly, a talk facility may have several chat rooms, each with different names. So you might, for example, enter the "Zen room". Once you do, you can see all the conversations going on in that room.

Public Events

Some talk facilities have special events in which anyone can gather in a particular chat room and listen to a guest speaker. (That room might be called the "Auditorium".) For example, there might be a scheduled time at which Bill Gates joins a particular chat room to give a speech on "Windows and Personal Enlightenment". During the speech (which Bill would type on his computer), everybody in the room could read what he was saying. Afterwards, there might be questions and answers. As people asked questions and Bill answered them, everyone could see what was happening.

Privacy

You can control whether or not a conversation is private or whether other people can join. If you are participating in a group conversation, you can initiate a temporary, private connection with one of the people (and tell secrets about everyone else). Generally speaking, if you are in a chat room, your conversation is not private.

White Pages Directory

Some talk systems keep a public directory in which you can look up someone's name (like a telephone book without the phone numbers). Once you find a name, you can send a request to that person to see if he or she wants to talk to you. Aside from the name, you may also see an email address, a link to the person's home page and other information.

A Personal Address Book

Most talk programs allow you to create your own personal address book. You can save the names and addresses of your friends on the Net, making changes whenever you want. To initiate a talk connection, all you have to do is display your address book and select the person to whom you want to talk.

Embedded HTML

HTML (Hypertext Markup Language) is the descriptive language used to build web pages (see Chapter 11). Some web-based talk facilities let you type HTML instructions. When these instructions are "read" by someone's browser, they will be interpreted as valid HTML. For example, as you are talking, you can send a link to your home page (or any web page), display buttons, make words blink, and so on. Of course, to do this, you have to know a little HTML. (It won't be long, though, until HTML is taught in schools as a second language.)

Sounds

You can send a sound effect to the person to whom you are talking. For example, you might send a laugh, a sigh, applause, and so on. Aside from pre-installed sounds, you can use sounds of your own. The other day, I was talking to someone in Sweden and he played a sound of a California surfer dude saying, "Okay, so I'll see ya later, huh?" I liked what I heard, so my Swedish friend used the file transfer facility to send me the actual sound file. I then registered the sound with my own talk program, so I could use it myself.

Pictures

You can use a picture to identify yourself. When people talk to you, they will see the picture. Some people use a photo of themselves, although this is not necessary: the picture can be any GIF or JPEG file. Some talk facilities have a library of pictures for you to choose from, if you don't have one of your own.

Video

There is a system called CU-SeeMe that allows you to send video images to another person on the Net. The video shows motion, although it is not of high quality like a television.

File Transfer

You can send a file to someone while you are talking without having to stop talking or without using a separate file transfer or mail program.

Answering Machine

If you are busy and do not wish to be disturbed, you can have your talk program send a specific message to people when they try to talk to you. When you want to start talking again, you can turn off the answering machine and take the calls yourself.

Remote Control

While you are talking to someone, you can control a program that is running on his or her computer. This will work for more than one person. For example, you could take control of someone's web browser and show them specific web pages. Or one person might take a group of people on a guided tour of the Web.

How Does It All Work?

Like all Internet services, talking on the Internet is based on a client/server system. There are two ways in which such systems are organized.

One design uses a central talk server, somewhere on the Net. Each person who uses the service runs a talk client on his own computer. The individual clients communicate by sending messages to and from the server.

For example, say you are using a centralized system to talk to a friend named Norman Sweetcream. Each time you type a line of text, your client program sends it to the server. The server relays the text to Norman's client program, which displays your message on his screen. When Norman types a line of text, his client sends it to the central server, which then sends the text to your client to display. In other words, everything you and Norman type goes through the server.

The other basic design connects people directly without using a central server. When you talk with Norman, everything you type goes right to his computer. When he responds, his messages go directly to your computer. With such systems, the program you run on your computer must be more than a talk client: It must act as both a client *and* a server.

There are advantages and disadvantages with both systems. The most obvious advantage of a decentralized system is there are no middlemen to slow down the talking. When everything has to go through a central server, things bog down and there is often a noticeable delay in the conversation.

In addition, with a decentralized system, no one has to maintain a central server. After all, why should someone (or some organization) have to pay for the hardware, software and administration time to support a talk server, just so people on the Internet can talk for free? Moreover, when someone *does* offer such a service to the Net, you must play by his or her rules or you are out.

Finally, a decentralized system is more dependable. When a central server goes down, everything stops. Such servers are attractive targets for hackers: talented programmers, who love to see how much mischief they can cause. From time to time, you will see a notice that the administrator has removed the "such and such feature" from a central server, because hackers were using it to cause trouble.

Centralized systems do have two big advantages, however. First, because the talk server is centralized, it can be more elaborate and offer more services than a server program you would run on your own computer. For example, a centralized server might create an imaginary environment (such as a mud) which the users can explore as they talk.

Second, a centralized system can provide a place for people to gather and to find one another. When you use IRC (Internet Relay Chat), for example, you will find many groups of people talking about different topics. Similarly, many people like to use muds because it allows them to meet new people and to hang out with their friends. When you use a decentralized system, there is no easy way to join a group of people or to just hang out.

What Do You Need to Use a Talk Facility?

To use an Internet talk facility, you need several things. First, your Internet connection should be either PPP or a direct network connection. If you use a shell account you will not be able to access some of the talk facilities. You will be able to use the Unix **`talk`** program (see Chapter 26), IRC (Chapter 27) and Muds (Chapter 28), but not much else. (For more information about Internet connections, see Chapter 5.)

Second, your Internet connection should be as fast as possible, especially if you want to use a voice facility. Thus, do not buy a modem that is slower than 28,800 bits per second. ISDN is even better. (See Chapter 4.)

Third, your computer should be capable of playing sounds. With a Macintosh, this capability is built in. With a PC, you must have a sound card or the equivalent. In addition, if you want to talk using voice, your computer must have a microphone.

Finally, you need the appropriate software. For all the talk systems I describe in this book, the software is available for free on the Net.

Types of Internet Talk Facilities

In this section, I will take you on a tour of some of the talk facilities you will encounter on the Internet. All of these are fun and useful. When you get a chance, experiment and see which talk facilities you like best.

The Unix `talk` Program

The oldest of the Internet talk facilities is a program named **`talk`** that runs under Unix. The **`talk`** program provides a basic, text-based talk facility. You can type messages back and forth to another person, and all the conversations are private.

The **`talk`** program allows you to talk with anyone else on the Internet, as long as they have a **`talk`** program. Virtually every Unix computer on the Internet has a **`talk`** program, so if you have a shell account, you can use **`talk`**. If you do not have a shell account (that is, if you use PPP or a direct network connection), you need to find a client for your own computer that is compatible with Unix **`talk`**.

When you use **`talk`** with a shell account, both the client and server run on the Unix computer. Thus, when you use a compatible program on your PC or Macintosh, the program must act as both a client and a server. Such programs are available for free on the Net.

The **`talk`** program is an important Internet service, and I discuss it in detail in Chapter 26.

INTERNET RESOURCE *Look in the catalog under Internet Resources for Talking: **`talk`** Clients*

Internet Relay Chat (IRC)

IRC is a huge Internet-wide system which supports a large number of group conversations. It is text-based and uses a number of central talk servers around the Net to coordinate the flow of data. IRC is a lot of fun and is a great place to meet people. To use IRC you need an IRC client for your computer. There are clients for all types of computers, including Unix, so you can use IRC with a shell account.

I discuss IRC in detail in Chapter 27.

INTERNET RESOURCE *Look in the catalog under Internet Resources for Talking: IRC Clients*

Muds

A mud is an imaginary environment in which you can talk to people and meet new friends. Moreover, some muds are set up as fantasy worlds, where you travel around and have adventures while assuming the role of an imaginary character. Many such muds are complex and require a long time to master.

You can access a mud by using a plain telnet client (see Chapter 23). However, there are specialized mud clients available as well. If you get into serious mudding, get yourself a serious mud client.

Muds can be a lot of fun and I encourage you to try a few. Many people tend to think of the Internet as being mostly the Web, but that is not true at all. If you haven't experienced a mud, you are missing out on one of life's most interesting experiences.

I discuss muds in detail in Chapter 28.

INTERNET RESOURCE *Look in the catalog under Internet Resources for*
Mud Clients
Telnet Clients

Web-Based Chat Rooms

The Web is huge, and there are a lot of talk facilities that make use of its size and popularity. To use these facilities you do not need a special talk client. All you need to do is connect to a centralized talk server, and your web browser will act as your talk client.

There are a lot of chat rooms on the Web, and they are worth checking out. However, there are some special considerations you should understand, which I discuss in detail in the next section.

INTERNET RESOURCE *Look in the catalog under Internet Resources for*
Talking: Web-based Talk

Bulletin Board Systems (BBSs)

Before the Internet became popular, there were a large number of Bulletin Board Systems (BBSs) that offered dial-up online services. Today, some of these BBSs, along with many new ones, have moved to the Internet. Many BBSs are run by an individual, although some are large enough to require a group of administrators. In general, the person who runs a BBS is called the SYSOP, and the personality of an individual BBS very much depends on the sysop.

Most BBSs allow you to send messages to people who are logged in. This is a good way to talk with people and meet new friends. There are usually ongoing discussion groups to which you can post messages as well as read the messages left by others. In addition, all BBSs have archives from which you can download files or to which you can contribute files of your own.

There are general-purpose BBSs, as well as BBSs devoted to just about any topic you can imagine: meeting people, religion, hobbies, corporate and government information, X-rated talk and resources, and so on.

To access a BBS over the Internet, you use telnet (see Chapter 23). Before you can use most BBSs, you will have to register and ask for an account. Many BBSs are free,

although some of them do charge a membership fee (especially the ones with the X-rated resources).

INTERNET RESOURCE *Look in the catalog under Internet Resources for* ***Talking: Bulletin Board Systems (BBSs)***

Talkers

A TALKER is a multiuser talk facility that is easy to use and is devoted primarily to conversation. You connect to a talker by using telnet (see Chapter 23). Once you do, you can talk to anyone else who happens to be connected at the same time. The word "talker" is a descriptive term. For example, some talkers are muds, while others are BBSs; there is no strict definition. If it's easy to use, and you connect in order to talk to other people, it is a talker.

INTERNET RESOURCE *Look in the catalog under Internet Resources for* ***Talking: Talkers***

Direct Talk Connections

Aside from the system based on the Unix `talk` program, there are other facilities that connect two people with a direct connection. Since such systems do not depend on a central server, the communication is fast and private. Some programs allow you to make connections between more than one person at a time to have a group conversation. The other people can be in the next room or on the other side of the world.

One program I particularly like is named Powwow. It not only supports text but, if you have the right hardware, voice as well. I use Powwow to send messages to my chief researcher, even though she often works in a room just down the hall. Such systems are great when you are participating in a telephone conference call. You can type secret messages back and forth to one of the people on the call without everyone else seeing what you are saying.

The Powwow system has a nice feature that is common to other talk facilities. There is a white pages server to which you can connect to find lists of people who use the same program. You can then choose someone on the list and see if they want to talk with you. In addition, while you are talking, one person can take control of the other people's web browsers (with permission, of course) and guide the other people on a tour of the Web.

INTERNET RESOURCE *Look in the catalog under Internet Resources for* ***Talking: Direct Connections***

Telephone-Like Systems

There are a number of voice-only talk facilities that are designed to work like a telephone. In order to use the system, you must have a PPP or a direct network connection (not a shell account). You run a special program on your computer, which acts as both a client and server. Using this program, you can talk to anyone else on the Net who is running the same program. (Of course, your computer needs a microphone and sound capabilities.)

How do you find someone to talk to? There are two ways. First, you can specify the exact address of someone's computer. If the computer name in the mail address doesn't work, you can always use a numeric IP address (see Chapter 6). However, you must know this address in advance, and you must be sure the person to whom you want to talk has his telephone program running, waiting for a connection.

Second, some systems have a number of special servers on the Net (maintained by the company that sells the telephone program). You can connect to a server and tell it you are ready to talk. At the same time, you can see all the other people who are ready to talk. To talk with someone, you simply select their name from the list on the server. The server will send a signal to that person. If he agrees to talk with you, a direct connection will be made between your computer and his. From that point on, you are not using the server (and everything you say is private).

Of course, since all of this happens on the Net, you do not have to pay long distance phone charges. So why doesn't everyone use the Net instead of a telephone? There are two reasons. First, the voice quality is not as good as a real telephone. Second, you can't call someone unless he has an Internet connection, a computer with a microphone and sound capability, and the same telephone program.

However, if you like talking to people for fun, try one of these systems. In most cases, you can use the software for free in a limited way. (For example, your conversations may be limited to one minute.) If you have someone to whom you would like to talk regularly for a long time, you can pay a small fee to the company and upgrade your program, so it allows you to talk as long as you want.

INTERNET RESOURCE *Look in the catalog under Internet Resources for* ***Talking: Telephone-like Systems***

Video Conferencing

A video conferencing system allows you to send live video and sound over the Internet. The most important such system is called CU-SeeMe. (The name is a pun. The "CU" stands for Cornell University where the original program development was done. However, when you say the name out loud, it sounds like "see you, see me".)

CU-SeeMe will run on either a Macintosh or a PC using Windows as long as you have a PPP or direct network connection (not a shell account). To watch video, all you need is the software, which is available for free on the Net. To broadcast, you need a video camera, a digitizing board and the appropriate video software.

When you watch, you will see actual video. (See Figure 25-3.) However, the quality is not all that good. (It looks like the video broadcasts from the space shuttle.) Still, it is exciting to watch people from all over the world. Although you need special equipment to send video, it is not that expensive, and there are a surprising number of people broadcasting over the Net.

By itself, the CU-SeeMe software will enable you to establish a video link between any two computers. However, by connecting to a "reflector" you can watch or broadcast to more than one site at a time.

One word of warning: A lot of schools use CU-SeeMe to let the kids talk to one another. Please remember, though, there is no censorship on the Internet. While I was working on this chapter, I saw more than one site broadcasting naked people performing quasi-sexual acts. Depending on your outlook, this could be either erotic or immoral. What I want you to appreciate is that the Internet is an open forum, and the only effective way to censor something is to control what comes into your own computer.

Web-Based Talk Facilities

The Web was not designed to allow individual users to communicate, so the talk facilities had to be fitted into the existing web client/server system. Because of this, you will often find web-based talking to be less smooth than other systems. When you

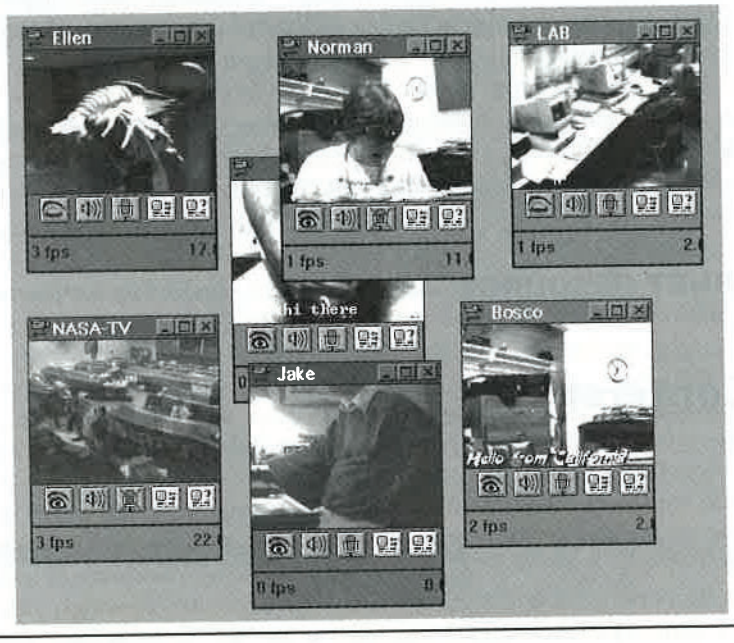

Figure 25-3. *Live video over the net, using CU-SeeMe*

talk with someone, you would like to see the words as the person is typing. With a web-based talk facility, this is not possible. Here is why.

Within the Web, you use a talk facility by connecting to a specific web page that represents a chat room. Whenever you load that page, you will see whatever messages people have typed in the last few minutes. An example of this system is shown in Figure 25-4.

How do you send a message of your own? On the Web, there is only one way to type information and send it to a server. The server must send you a web page that contains a special input area called a "form" (see Chapter 10). You type information into the form and then submit it, either by pressing the ENTER key or by clicking on a button. (Notice the input form in Figure 25-4.)

This means the information you type is not sent to the server until you submit the form. At that time, everything you typed into the form is sent all at once. The talk server receives your input in chunks.

Similarly, there is no way for a web server to send you information one character or one word at a time. All the server can do is wait for you to reload a web page and then send you updated information.

When you put this all together, it explains why talking on the Web is not a smooth process. Say you are talking to a friend. You must type your message into a form, and nothing will be sent to the server until you submit the form. At the other end, your

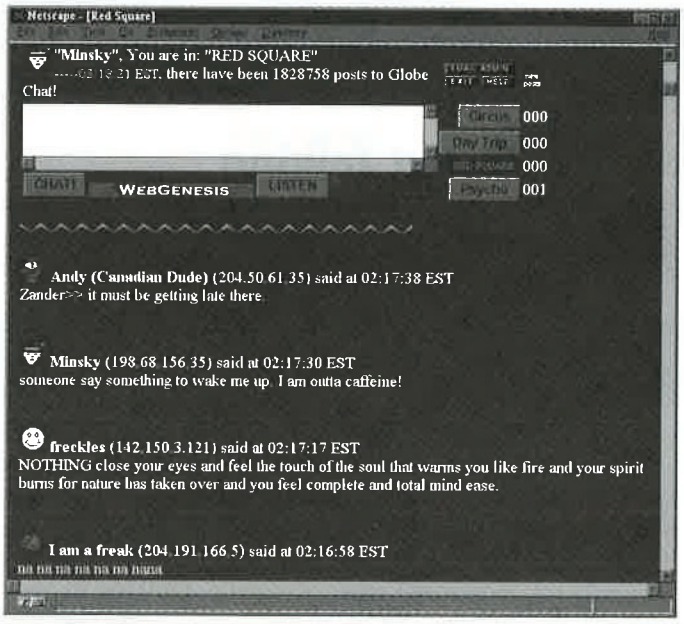

Figure 25-4. *A typical web-based chat room*

friend will not see anything until he or she reloads the web page. It is only at then that the server will be able to send any new information.

(Some talk systems mitigate this problem by using server push [see Chapter 10]. This allows the web server to update your web page automatically at regular intervals. With such systems, you can sometimes specify the interval, so you can control the time between updates.)

Thus, web-based talking is not smooth for several reasons. First, all messages must pass through a central server. Second, a message is sent to the server as one large chunk, not one character at a time. Third, you will not see any new messages until you reload the chat room web page. Finally, everyone in the room can talk at the same time. Thus, each time you reload your web page, you will see *all* the messages that have been sent by everyone in the room, since the last time you reloaded the page. As you might imagine, this can be a tad confusing.

Still, after a while, you will get used to web-based talking. It is only a matter of waiting for certain cells in your brain to change appropriately. Moreover, many of the servers are quite elaborate, offering an interesting variety of talk-related resources such as specialized chat rooms, public events, and so on. So, in spite of all the limitations, web-based talk facilities are worth investigating.

HINT

Most web-based chat systems place the newer messages at the top of the web page. Thus, to look at the messages in chronological order, you must read from the bottom up.

Guidelines for Text-based Talking

Having a text-based conversation on the Internet combines the immediacy of a face-to-face encounter with the verbosity of written language. For this reason, you will notice two significant differences between talking by computer and talking by mouth.

First, talking by computer is a lot slower, because almost everybody can speak faster than they can type (not that this is always an advantage). Second, because you are not looking at the other person, the customary nuances and inflections that comprise so much of the non-verbal aspects of communication are missing.

We make up for these limitations in two ways. First, there are a number of abbreviations that are commonly used to speed things up. Some of these are shown in Figure 25-5. These abbreviations are the spiritual descendants of the Morse-code conventions used by ham radio operators since the 1920s.

The second important convention recognizes that it is easy to offend someone (or at least have your words misconstrued) when you are not talking in person. Over the computer, we are unable to use subtle body language to indicate that what we just said was debonair and ironic, and not really as dumb and insensitive as it looks.

Abbreviation	Meaning
BBL	be back later
BCNU	be seeing you
BRB	be right back
BTW	by the way
BYE	goodbye (I am ready to stop talking)
BYE?	goodbye? (are you ready to stop talking?)
CU	see you (goodbye)
CUL	see you later
CUL8R	see you later
F2F	face to face
FWIW	for what it's worth
IMHO	in my humble opinion
JAM	just a minute
L8R	later
LOL	laugh out loud
OBTW	oh, by the way
R U THERE?	are you there?
ROTF	rolling on the floor
ROTFL	rolling on the floor laughing
SEC...	wait a second
TTFN	ta-ta for now
TTYL	talk to you later
WRT	with respect to

Figure 25-5. *Common communication abbreviations*

Thus, when there is any doubt as to whether or not the other person might be offended, it is a good idea to use a smiley to show that you are just kidding. A smiley is a little face that we draw with punctuation characters. Here is the basic smiley:

```
:-)
```

(To see the face, tilt your head to the left.) You can use a smiley whenever you say something that might bother the other person, for example:

```
I'd feel the same way as you, if I had a
head that looked like a ripe eggplant :-)
```

If you want to see a few more smileys, look at Figure 14-5 in Chapter 14.

It is also important to become aware of what is private and what is not. With some talk systems, it is possible for people to sit quietly and watch what other people are saying. Such people are called LURKERS. Just because you see only one person talking does not mean that no one else is watching.

HINT

When in doubt, assume your conversation is not private.

Some talk programs have a way for you to talk with a group, but break off temporarily to have a private conversation with just one person. If you have never used this facility, be sure to test it. It is a good idea to be completely sure that you understand how the privacy option works before you say anything you will regret.

A further consideration has to do with how the words you type are sent to the other person. With some systems, what you type is not sent until you press the ENTER key or click on a button. In such cases, you can backspace to make corrections before you send the text. On other systems, the message is sent, one character at a time, as you type it. (This is the case with the Unix `talk` program.) Thus, if you accidentally say something insulting, by the time you backspace and change it, the other person will already have seen it.

If the network or the talk server slows down, you may notice replies are delayed and become unsynchronized, leading to various amusing and incoherent situations. In such cases, we refer to the delay as LAG. Thus, you might see someone say, "There was so much lag, I thought you had forgotten all about me."

HINT

For possibly the first time in your life, spelling does not count.

Because typing is so much slower than talking, it is perfectly acceptable to type as fast as you can and not worry about spelling mistakes. Nothing is worse than watching someone laboriously backspacing and correcting each word. Just go ahead and type and don't worry. Almost evrWything is underftandabll with kust a few worng lettrs and. no doubt,m the other pweosn will get the g9ist of thwta you are syaing.

Chapter Twenty-Six

Using Talk from a Shell Account: `talk`, `ytalk`

HECKMAN
BINDERY INC.
9-147-973
N. MANCHESTER,
INDIANA 46962

Internet/PC/Macintosh and Unix

Skill Level Guide

- ☐ Beginner
- ☐ Intermediate
- ☐ Advanced
- ☑ For Every User

Everything You Need to Know to Use the Internet

The Internet. It's huge and it's growing. How can anyone keep up with all the changes? Let Harley Hahn, the best-selling Internet author of all time, guide you through the latest developments of the Net, the worldwide information network. Harley Hahn's **The Internet Complete Reference, Second Edition** is completely revised and rewritten, and contains everything you need to know about the Internet:

- What hardware and software you need to connect to the Internet
- How to use the Web, Usenet, gopher, and anonymous ftp
- How to communicate with people all over the world using electronic mail, web chat rooms, IRC, muds, and more.

With an emphasis on how to use Windows and Mac clients, **The Internet Complete Reference, Second Edition** also covers client programs for Unix shell account users: the **pine** mail program and **pico** text editor, the **lynx** Web browser, the **tin** newsreader, **ftp**, **telnet**, and much more.

The coverage of the Web is extensive, including:

- What the Web really is
- How to use the Web
- Web Pages and Home Pages
- Understanding URLs
- Links, Forms, and Image Maps
- Web directories and Search Engines

Whether you use a PPP, a shell account, or a direct Internet connection, **The Internet Complete Reference, Second Edition** is for you. With the most authoritative and extensive reference ever published, anyone can enjoy the Internet's limitless potential!

OSBORNE
Get Answers—Get Osborne
For Accuracy, Quality and Value

7 83254 02138 9

ISBN 0-07-882138-X

90000

9 780078 821387

$32.95 US