

ULTRABASIC GUIDE TO THE INTERNET

FOR HUMANITIES USERS AT UCSB

Version 1.0 (Beta)

By Alan Liu

English Dept.

Dedicated to all savage humanists

©1994 Alan Liu

CONTENTS

Preface.....	i
Acknowledgements.....	i
Conventions Used in this Manual.....	ii

PART I: BASIC SKILLS

CHAPTER 1: ACCESS AND LOGON/LOGOFF

1.1 Access to the Internet	1
1.2 Logging On.....	2
1.3 Logging Off.....	8

CHAPTER 2: DOWNLOADING/UPLOADING

2.1 Overview.....	9
2.2 From Faculty-Office and Department Computers Hardwired to Humanitas.....	9
2.3 From Microcomputer Laboratory Computers	11
2.4 From Home Personal Computers By Modem	12

CHAPTER 3: GETTING OUT OF TROUBLE

3.1 How to Exit, Escape, Interrupt	17
3.2 How to "Kill"	18
3.3 How to Ask for Online Help.....	18

PART II: SURVIVING UNIX

CHAPTER 4: UNIX OVERVIEW

4.1 The Idea of Unix.....	23
4.2 Your Assigned Storage Space	23
4.3 Directories and Files.....	24
4.4 Tree Structure	25

CHAPTER 5: DIRECTORY MANAGEMENT COMMANDS

5.1 Where Are You? (<i>pwd</i>).....	27
5.2 Listing the Contents of a Directory (<i>ls</i>).....	27
5.3 Creating or Making Directories (<i>mkdir</i>).....	29
5.4 Changing Directories (<i>cd</i>).....	29
5.5 Renaming Directories (<i>mv</i>).....	30
5.6 Removing or Deleting Directories (<i>rmdir</i>).....	30

CHAPTER 6: FILE MANAGEMENT COMMANDS

6.1 Viewing Contents of Files (<i>more</i> or <i>pg</i>).....	31
6.2 Creating and Editing Files (using <i>Pico</i>).....	33
6.3 Renaming and Moving Files (<i>mv</i>).....	35
6.4 Copying Files (<i>cp</i>).....	36
6.5 Concatenating Files (<i>cat</i>).....	37
6.6 Deleting or Removing Files (<i>rm</i>).....	37
6.7 Uuencoded, Compressed, and Archived Files	37

CHAPTER 7: DIRECTORY AND FILE PERMISSIONS

7.1 "Ownership" in Unix	41
7.2 Changing Permissions (<i>chmod</i>)	42

PART III: THE INTERNET

CHAPTER 8: OVERVIEW OF THE INTERNET

8.1 Capsule History.....	47
8.2 General Principles.....	48
8.3 The Interface	50

CHAPTER 9: E-MAIL (AND FINGER SEARCH ENGINE)

9.1 Pine E-Mail.....	53
9.2 Pine Main Menu Screen (Starting Pine)	53
9.3 Folder Index Screen (Seeing the List of Messages in Your Inbox)	55
9.4 Message Text Screen (Reading a Message)	56
9.5 Compose Message Screen (Writing and Sending a Message)	59
9.6 Folder List Screen (Managing Folders).....	62
9.7 Addressbook Screen (Keeping a Contact List)	63
9.8 Finger Search Engine	64

CHAPTER 10: LIST SERVERS ("MAILING LISTS")

10.1 The List Server Concept.....	67
10.2 Finding Interesting Lists.....	67
10.3 Communicating with the Administrative Address	69
10.4 Communicating with the List-Title Address.....	71

CHAPTER 11: FTP (AND ARCHIE & WAIS SEARCH ENGINES)

11.1 The FTP Concept	73
11.2 Anonymous FTP Logon/Logoff.....	73
11.3 Browsing Remote FTP-Servers.....	74
11.4 Downloading Files from Remote FTP-Servers.....	79
11.5 Issuing Local Commands While in a Remote FTP Session.....	80
11.6 Archie and Wais Search Engines	80

CHAPTER 12: ANONYMOUS TELNET (AND HYTELNET SEARCH ENGINE)

12.1 The Telnet Concept.....	89
12.2 Opening and Closing a Telnet Session.....	89

12.3 Hytelnet Search Engine.....	90
----------------------------------	----

CHAPTER 13: USENET NEWSGROUPS (nn NEWSREADER)

13.1 The Usenet Concept.....	93
13.2 Subscribing to the News: The ".newsrc" File	95
13.3 Starting the nn Newsreader and Pointing it to a Newsgroup	96
13.4 Selecting Messages (the nn Select Messages Screen)	97
13.5 Reading Messages.....	98
13.6 Downloading Messages.....	100
13.7 Posting and Responding to Messages.....	102
13.8 Newsgroup Housekeeping (Clearing, Catching Up, Unsubscribing, Subscribing)	104
13.9 Changing Newsgroups and Quitting nn.....	105

CHAPTER 14: GOPHER (AND VERONICA SEARCH ENGINE)

14.1: The Gopher Concept.....	107
14.2: Starting (and Quitting) Gopher	107
14.3: Navigating Gopher Menus.....	110
14.4: Reading and Downloading Files, Etc	111
14.5: Bookmarks.....	112
14.6 Veronica Search Engine	113

CHAPTER 15: WORLD WIDE WEB (LYNX AND MOSAIC BROWSERS)117

[Incomplete]

APPENDICES

APPENDIX 1: MODEMS AND COMM PROGRAMS: A PRIMER.....	119
---	-----

APPENDIX 2: ALAN LIU'S for_all_hands DIRECTORY	121
--	-----

*Index of Key Terms (Double-Underlined Terms) Used in a Defining Context*¹²³

PREFACE

There are many printed guides to the Internet in bookstores as well as online guides distributed on the Net itself. The purpose of the Ultrabasic Guide is to complement this body of materials by providing a concise introduction specifically for the humanities community at UCSB. Included are instructions on access, downloading/uploading, Unix, and the Internet as they work here on the UCSB Humanitas server. "Snapshots" of actual online sessions accompany the instructions.

Since the author is an English professor and not a computer specialist, there is bound to be some technical advice here that a maestro would consider inexact, incomplete, inefficient, or worse. To apply Claude Lévi-Strauss's concept: this is a "bricoleur's" guide to the Internet, not an "engineer's." It is for all those who approach the computer with a "savage mind." (In Lévi-Strauss's anthropology, "savage" thought is akin to the mentality of the Western bricoleur or jack-of-all-trades who, when confronted with a problem, goes into the garage, digs into a bin full of odds and ends, and concocts some Rube-Goldberg-like gizmo that does the trick whether or not it

☛ Except for the Mac-oriented sections focusing on procedures at the UCSB Microcomputer Laboratory, this manual takes the viewpoint of an IBM-PC user with an account on the Humanitas server computer (running the default Unix C-Shell). However, most Unix and Internet practices are similar regardless of whether you are working through an IBM or Mac "front end," and regardless of which campus server (Humanitas, MCL, Alishaw, UCSBUXA, etc.) you have your account on. Mac users and those with accounts on other servers should be able to use all parts of this manual with only slight adaptations.

is the most "systematic" solution. Such is not science in the modern acceptation, Lévi-Strauss argues, but nevertheless constitutes its own intricate, deep, and rigorous way of knowing the world.)

ACKNOWLEDGMENTS

Thanks to Terri Jo Ortega of the English Department (System Administrator of Humanitas) for answers to many questions and to the faculty liaison folks at the Microcomputer Laboratory for help with MCL-specific topics. Thanks also to Francis Steen and Vincent Willoughby for tips and online fellowship.

I have consulted a number of books and journals, most importantly:

Peter Norton and Harley Hahn, Peter Norton's Guide to Unix (New York: Bantam, 1991)

Ben Hunsburger, Unix Quick Reference (Indianapolis, IN: Que, 1994)

Gail Anderson and Paul Anderson, The Unix C Shell Field Guide (Englewood Cliffs, NJ: Prentice-Hall, 1986)

Ed Krol, The Whole Internet : User's Guide & Catalog, 2nd ed. (Sebastopol, CA: O'Reilly & Associates, 1994)

Daniel P. Dern, The Internet Guide for New Users (New York: McGraw-Hill, 1994)

Harley Hahn and Rick Stout, The Internet Complete Reference (Berkeley, CA: Osborne McGraw-Hill, 1994)

Internet World: The Magazine for Internet Users, various issues

Wired Magazine, various issues

In addition, several RFC's and other information files available through the Internet have been helpful. Special acknowledgement goes to a predecessor work: Albert S. Lindemann's Basic E-Mail at UCSB (For Users of HCFMAIL, MCL, ALISHAW, and UCSBUXA) (UCSB: Humanities Computing Facility, 1992).

This is a "beta" release of the Ultrabasic Guide. Please send suggestions and bug-reports to the author at ayliu@humanitas.ucsb.edu. Certain features of the Internet are not covered in the present version of this manual, but may later be included if there is sufficient interest and/or hardware capability among humanities users (e.g., Internet Relay Chat, GUI as opposed to text interfaces to the World Wide Web).

CONVENTIONS USED IN THIS MANUAL

- ▶ Important terms are underscored by a double-underline when first used in a defining context. (Note: this feature is not yet fully implemented in version 1.0 of this manual.)
- ▶ Instructions for commands are printed in **boldface** and assume that the <Return> key will be pressed afterwards (for example, "ls" means, "type 'ls' followed by <Return>").
- ▶ Function keys are enclosed by angle-brackets (e.g., <Return> = "return," <Spc> = "spacebar").
- ▶ Control-Key combinations are represented as "^Key" (e.g., "^C" means press the Control and C keys simultaneously).
- ▶ Contrary to normal punctuation rules, I place periods and commas outside quotation marks where there may otherwise be confusion.

For example, this sentence ends in a filename that does not include a period: "filename".

- ▶ The term "process" is a catch-all referring to any command, routine, or program running in the Unix environment.
- ▶ "humanitas>" is the prompt you see when logged onto the Humanitas server computer. For undergraduate users working at the Microcomputing Laboratory and the MCL server, the equivalent prompt is some variation on the following: "yourname@mcl21%".
- ▶ At various points in the following instructions, I reproduce "snapshots" of actual online sessions. These are framed in a box and embellished in two ways. First, commands the user needs to type are in boldface. For example, here the user types "ls" at the humanitas> prompt followed by <Return>:

```
humanitas> ls
```

Secondly, square brackets indicate an argument or parameter the user needs to customize. For example:

```
humanitas> rm [filename]
```

Here, the user supplies a filename for the **rm** (delete file) command to act upon.

- ▶ When describing certain programs with a menu interface (e.g., nn, Lynx), I refer to the "highlight bar." On some computers, however, the highlight bar is represented differently (e.g., as a phrase or line in boldface characters).

PART I

BASIC SKILLS

ACCESS & LOGON,
DOWNLOADING/UPLOADING,
GETTING HELP & GETTING OUT OF
TROUBLE

CHAPTER 1:

ACCESS AND LOGON/LOGOFF


1.1: ACCESS TO THE INTERNET

Humanities Faculty and Graduate Students: All UCSB community members have free access to the Internet through University-provided accounts. Humanities faculty and graduate students are assigned Humanitas accounts--meaning that they access the Internet through the Humanitas server computer located in the English Dept. office. (Some humanities faculty and graduate students have accounts on the UCSBUXA server, which means that they are technically part of a network called Bitnet rather than the Internet proper. But Bitnet at UCSB is set up to give transparent access to the full range of Internet services.) To use your Humanitas account, you can dial in via modem from your home computer or use a faculty-office or department computer hardwired to the Humanitas server (through the campus "backbone"). If you don't already have a Humanitas account, call Terri Jo Ortega in the English office at 893-2119 to set one up.

An account user to choose a (or "userid") and a Your username is three- to eight-letter combination based on name--e.g., "ayliu". users have a numeric prefix--e.g.,

Your password must the following rules of


- ▶ It should be seven or longer.
- ▶ It should use both numbers.
- ▶ It should not be you in any (e.g., don't use initials, your security number, "Wordsworth" if romantics
- ▶ It should not be a likely to be on of the peculiar the population computers ("klingson," "ferengi," and "worf" are out).
- ▶ It may include some alternation of lower-case and upper-case characters to increase security. The UNIX operating system is case sensitive.

 Use the `passwd` command periodically to change your password. Password security is important because of the number and sheer scale of past attempts to suborn legitimate Internet access (witness the famous "worm" virus that some years ago propagated on the Internet by automatically guessing the password of users). Humanities users are often casual about security because they do not believe they have any data likely to be raided. The sad truth is that outside hackers often could care less about your data; they just want your access to the Internet. You would be amazed how much harm can be done by a nefarious hacker acting in your name. Consider that the White House is now on the Internet. One person has already been arrested for sending an assassination threat by e-mail.

requires the username password. usually a your (UCSBUXA "6500ayl".) accord with thumb: characters letters and related to obvious way your social or you are a specialist). phrase the minds subset of who hack

Undergraduates: As of academic year 1994-95, all undergraduates will automatically be issued an Internet account

(sometimes called an "e-mail account") complete with username and password. The account is on the Microcomputer Laboratory or "MCL" server computer. Students who cannot access this server on their own personal computers via modem can use the Mac or IBM computers at the Microcomputer Laboratory. The Lab has both "open times" (when anyone can get on the machines on a first-come first-served basis) and "reserved times" (when students enrolled in certain courses have priority). If you do not know your username and password or have other questions, stop by the Lab (1507-1530 Phelps) or call 893-3002. The schedule of open and reserved times each quarter is available by phone at 893-8414.

 Users can complement their UCSB Internet accounts with private, commercially-provided accounts. For example, such services as America OnLine or Delphi offer Internet access among their other resources at a rate of \$10-\$20 per month for up to 5-10 online hours (extra time is charged at an hourly rate). NetCom, which specializes in Internet access, does not currently have a local-access phone number for Santa Barbara (but they plan such a number in the future). To date, few if any commercial services are quite as good for using the Internet as our own UCSB connection. For example, some commercial services offer only bare-bones news readers for Usenet, charge a premium for business-hour usage, charge extra for downloading more than a set amount per


So why consider a commercial account? (a) You like the other resources of the service; (b) you want a Windows interface or other amenities (America OnLine, for example, has a Windows front-end); (c) you want redundant Internet access for those times when your UCSB server is "down"; (d) you want to keep your university-related and "private" online activity separate (ethically, of course, you should be using your UCSB account for mostly official purposes; and, practically, you may want to keep some online activity separate from your university account for security or legal reasons).

1.2: LOGGING ON

To get on the Internet, you must logon to the campus server computer on which you have your account (this manual assumes by default that you are using the Humanitas server). The logon process is best conceived as a two-step routine in which you first connect your personal computer to the server and then open your account on that server. The former step, here called "local level," varies depending on your local, physical link to the server (e.g., "hardwired" or modem). The latter step, or "server level," is standard for all users of a particular server. Once you are logged in, your server automatically connects you to the campus net and the global Internet as needed.

Below are logon instructions for (A) faculty and graduate students working on a computer hardwired to the

Humanitas server; (B) under-graduates working at the Micro-computer Laboratory on a Mac hardwired to the MCL server (the IBM machines at the lab are not covered in this manual); and (C) any user dialing their server via modem.

 Note: users working via modem from their home pc can create "scripts" in a communications program that will automate the whole logon process from start to finish. But it is useful to know the manual logon process in the event that the automated process fails. In any

A. Faculty and Graduate Students Hardwired to the Humanitas Server:

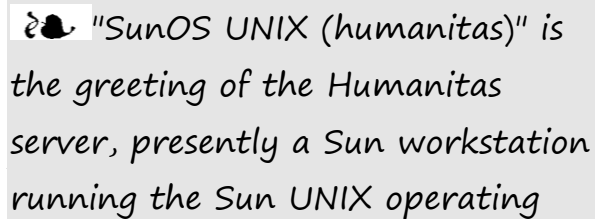
Local Level: The steps needed to connect your personal computer to the server vary depending on how your department has configured its faculty and graduate-student computers. The routine for English Dept. faculty--whose office computers are part of a LAN or local area network--is as follows:

- From the Novell menu that appears when you have logged onto the English Department LAN, choose "F" for "E-Mail." (E-mail is just one feature of the Internet.)
- The message, "SunOS UNIX (humanitas)", will appear followed by "login:", indicating that you are ready to move on to the server-level logon (see below).

The equivalent routine for graduate students working in the English department's computing room is as follows:

- From the opening menu on one of the two IBM computers dedicated to the Internet (as well as to Melvyl and Pegasus), choose the "GOHUMAN" option by typing "gohuman <Return>".
- The message, "SunOS UNIX (humanitas)", will appear followed by "login:", indicating that you are ready to move on to the server-level logon (see below).

Server Level: Once your personal computer is connected to the Humanitas server, which identifies itself as "SunOS UNIX (humanitas)", then the rest of the logon procedure is standard. This is the server-level logon:



"SunOS UNIX (humanitas)" is the greeting of the Humanitas server, presently a Sun workstation running the Sun UNIX operating

- When you see "login:", enter your username followed by <Return>.
- When you then see "password:", enter your password. (Note: to foil the person looking over your shoulder, the screen does not display your password as you type.)
- After various messages scroll by, you will see "TERM = (vt100)". This is the server asking if you want it to interpret commands from your particular keyboard according to VT100 emulation (a standard set of

key assignments inherited from the early days of computing). Go <Return> to accept VT100 as your default.

- The "humanitas>" prompt will then appear, indicating that you are logged on to the Humanitas server and ready to go.

The whole session at the server level on Humanitas--excluding the local level--looks like this:

```

SunOS UNIX (humanitas)

login: ayliu
Password: [password]
Last login: Fri Jul 15 21:23:47 from 128.111.224.251
SunOS Release 4.1.3_U1 (GENERIC) #3: Fri Jul 15 07:48:03 PDT 1994

*****WELCOME TO HUMANITAS*****

If you need assistance with your e-mail account, contact your departmental
computer support, or Terri Jo Ortega at 2119 or e-mail tortega@humanitas.

If you have general computer questions, please contact askhcf@humanitas.
You should receive a response in 24 hours.

You have new mail.
TERM = (vt100) <Return>

humanitas>

```

B. Undergraduates Working on a Mac LC at the Microcomputer Laboratory:

Local Level:

- ▶ When you sit down in front of one of the Mac LC computers, you may see a screen-saver display on the screen. If so, move the mouse in any direction to cancel the screen saver and display the real screen. You will see a button marked "Start." Push it (i.e., use the mouse to put the cursor over the button and double-click the left mouse button).
- ▶ The next screen to appear shows the image of a large file folder with many small folder icons inside. Double-click on the small folder marked "NCSA Telnet 2.5" (the version number may change in the future). This opens the program that is the Lab's interface to the Internet.
- ▶ When the NCSA Telnet 2.5 screen comes up, there is a menu bar across the top. Hit "File" to invoke the File drop-down menu. On the drop-down menu, hit "Open Connection".
- ▶ The Connections dialogue box will then open with the "session" option set to "mcl". If the option isn't set, enter mcl in the appropriate blank. Then hit the "Okay" button. (What you are doing is connecting your machine to the mcl server computer.) Note: on some computers at the Lab, this step is skipped entirely and you proceed directly to the server level.

Server Level:

- ▶ When you see "login:", enter your username followed by <Return>.
- ▶ When you next see "Password:", enter your password.
(Note: to foil the person looking over your shoulder, the screen does not display your password as you type.)
- ▶ After various messages scroll by, you will see "TERM = (vt100)". This is the server asking if you want it to interpret commands from your particular keyboard according to VT100 emulation (a standard set of key assignments inherited from the early days of computing). Go <Return> to accept VT100 as your default. (If this doesn't work, go ^C and start over.)
- ▶ The mcl prompt will then appear, indicating that you are logged on to the mcl server and ready to go. The prompt has the following format:
"[name]@mcl21%", where "[name]" is your user name and the number just before the "%" sign varies depending on which group of mcl users your account was assigned to.

C. From a Home Computer Via Modem:

Local Level:

- ▶ Configure your 2,400 or 14,400 baud modem, the communications port on your personal computer, your communications program, and your dialing session as instructed in the **Appendix** to this manual

on "Modems and Comm Programs: A Primer."

These settings need only be entered once. The phone number for the 2,400-baud modem bank on campus is 893-8400; that for the 14,400-baud modem bank is 893-3000. The session parameters are 7-bit, parity=none, stop bits=1, emulation=VT100. (See Appendix for details.)

- ▶ Dial Humanitas by using the appropriate commands in your comm program. The standard, manual dialing process is to enter ATDT on the communications screen of your comm program followed by the appropriate phone number; but all comm programs have means to automate and customize the dialing process. For example, some programs allow you to put the cursor over Humanitas (or some other item) on a menu and hit "d" for dial. Your modem will open the line, dial, and issue a set of whistles and hisses during which the "carrier wave" is established and a "handshake" accomplished between your local modem and the remote modem bank. When you see "CONNECT 2400" (or some other baud rate) on your screen, you are ready to proceed to the server-level phase as follows.

Server Level (including the link between the campus modem-bank and campus server):

2,400 Modem:

Until sometime in October, 1994, the logon for 2,400 modems dialing the 893-8400 number and

connecting to Humanitas is as follows (MCL users see a slightly different screen). Thereafter, a new logon process will be required. The new process will be identical with that already in place for the 14,400 modems (see under "14,400 Modem" below).

```
CONNECT 2400
<Return>

You may now enter Net/One commands
>
> connect gaucho
connecting...(8) 9475a3 success
<Return>

LANTRONIX ETS-16 Terminal Server Version V2.2/36(930609)

Type HELP at the 'Local_5>' prompt for assistance.

Local_5> telnet hcfmail
Telnet protocol emulation V2.2

SunOS UNIX (humanitas)

login: [your user name]
Password: [your password]
Last login: Fri Jul 15 21:23:47 from 128.111.224.251
SunOS Release 4.1.3_U1 (GENERIC) #3: Fri Jul 15 07:48:03 PDT 1994

*****WELCOME TO HUMANITAS*****

If you need assistance with your e-mail account, contact your departmental
computer support, or Terri Jo Ortega at 2119 or e-mail tortega@humanitas.

If you have general computer questions, please contact askhcf@humanitas.
You should receive a response in 24 hours.

You have new mail.
TERM = (vt100) <Return>

humanitas>
```

The "humanitas>" prompt means that you are in!

☛ Certain numbers in the logon sequence vary each time--e.g., "Local_5>" can be "Local_3>," "Local_8>," etc. Also, the system administrators sometimes make minor changes in the logon messages or prompts, necessitating adjustments in any automated script you have created. If your script gets stuck at any point, just take over manually. If things are "foobar" (slang in the Unix world for "f__ up beyond repair"), try hitting the "Break" key on your keyboard and restarting the sequence manually from the "connect" step. As a last resort, use your console program to

14,400 Modem:

Previous to Sept. 20, 1994, the logon sequence for 14,400 modems dialing the 893-3000 number and connecting to Humanitas was a variant of the process for 2,400 modems (as shown above). After that date, the high-speed modem bank was hooked up to a new "terminal server" necessitating a logon procedure as follows (MCL users see a slightly different screen).

Boldface indicates the commands you type:

CONNECT 14000

UCSB Communications Services high-speed modem pool.

To initiate a connection, please depress the Carriage Return key twice.

To report problems with these modems, please call our Trouble Desk at

893-4600 (7:30 a.m.-5:00 p.m. on University business days). For information

on modem configuration parameters and questions regarding this dial-in

modem pool, please call our Customer Relations section at 893-8700.

<Return>

<Return>

Welcome to the UCSB Communications Services Terminal Server

Searching for script file. Please wait...

Welcome to the UCSB Communications Services Terminal Server

The following commands are available:

Connect service < where service is a UCSB system providing telnet access
 Connect NET1 < connects to the Net/One network
 Help < provides additional help
 LOGout < disconnects

Once connected to a system, the <break> key returns the user to the terminal server command mode. Use the 'RESume' command to return to the session. The 'CLOse' command can be used to close a hung session. While in command mode the command 'SET SESSion PASSall' will set the session to allow binary data transfers. To prevent the <break> key from returning the user to the terminal server, issue the command 'SET BREak REMote'. Wylbur users may wish to use this command. Wylbur users may need to set 7-bit display if using emulators.

The command 'Connect NET1' connects to Net/One. Once connected to Net/One, proceed as usual. The Net/One 'QUIT' command returns to the terminal server. Comten IPC users should enter 'Connect NET1' and then 'C IPC' as before.

UCSB> connect hcfmail

Xyplex -O10- Session 1 to HCFMAIL established

SunOS UNIX (humanitas)

login: agliu

Password:

Last login: Tue Sep 27 09:34:44 from 128.111.253.101

SunOS Release 4.1.3_U1 (GENERIC) #4: Mon Jul 18 11:27:42 PDT 1994

*****WELCOME TO HUMANITAS*****

If you need assistance with your e-mail account, contact your departmental

computer support, or Terri Jo Ortega at 2119 or e-mail tortega@humanitas.

If you have general computer questions, please contact askhcf@humanitas.

You should receive a response in 24 hours.

You have new mail.

TERM = (vt100)

humanitas><Return>

The "humanitas>" prompt means that you are in!

☺ The change in dial-in logon procedure for 14,400 modems instituted on Sept. 20, 1994 (and scheduled for 2,400 modems sometime in October 1994) is the result of a campus hardware upgrade. Instead of connecting initially with the campus Net/One network, then with the "gaucho" machine, and finally via remote telnet with Humanitas, users now connect with a new "terminal server" (a special machine) capable of bypassing Net/One and starting a remote telnet session with Humanitas directly (and invisibly) when the user types "connect hcfmail" or "connect humanitas". Thus "connect hcfmail" now replaces the older, two-step process of "connect gaucho" and "telnet hcfmail". (On the concept of telnet, see §12.1.)

1.3: LOGGING OFF

To log off from the system, stop any process you are currently running (for example, "quit" from Pine e-mail) and then type "bye" at the humanitas> prompt. If you are working

on a campus computer hardwired to the server, then this may be all that is needed. In the case of modem connections, "bye" needs to be followed by a "logout" as in the following example. (You can instruct your communications program to "hangup" the phone anytime after you have issued the "logout" command without waiting for the rest of the scenario.) Here is the logoff process for a Humanitas modem connection (UCSBUXA users may also see some accounting information about computing time):

```
humanitas> bye
```

```
Xyplex -O11- Session 1 disconnected from HCFMAIL
```

```
UCSB> logout
```

```
Xyplex -O20- Logged out port 15 on server XO208DE at 27 Sep 1994 16:34:33
```

```
NO CARRIER
```

CHAPTER 2:

DOWNLOADING/UPLOADING

2.1: OVERVIEW

One of the single most useful skills you can learn is downloading/uploading files between the Humanitas server and whatever local computer

you are working on (whether in your office, your department's computer room, the MCL, or your home).

Downloading allows you to transfer files from your assigned storage space on the Humanitas server to your local machine's hard or floppy drive. (On managing Unix directories and files in your assigned server space, see Part II of this manual.) Uploading, inversely, allows you to

transmit files from your local machine (e.g., a document created in your favorite word processor) to your server storage space for dissemination to others via e-mail or other means.

👤 In the strange topography of cyberspace, the server computer sits high "above" as if at the top of a mountain. By contrast, your personal computer (a.k.a., the "local," "remote," or "client" machine) sits abjectly "below" (despite the fact that "service," of course, is supposed to connote the more humble position). Thus sending a file from the server to your personal computer is called "downloading," and the reverse is "uploading."

The procedure for downloading/uploading varies depending on whether you are working from a faculty-office or department computer hardwired to Humanitas, a Microcomputer Laboratory machine, or your home personal computer.

2.2: FROM FACULTY-OFFICE AND DEPARTMENT COMPUTERS HARDWIRED TO HUMANITAS

Different departments may have their machines set up differently. The main idea is that machines hardwired to Humanitas via the campus backbone can use the FTP (File Transfer Protocol) component of the Telnet program to transfer files back and forth to the server. By comparison with files transfers via modem, this method is very quick.

The following is the routine for English Dept. machines:

- ▶ At the humanitas> prompt, type "Alt-t". Press the "Alt" and "t" keys simultaneously, that is, to invoke the Telnet program. (Though you are working on a campus computer hardwired to Humanitas, this step is logically analogous to dialing in to Humanitas by modem and going "telnet hcfmail." In essence, you are opening a special kind of connection to the server on top of the connection you have already established.)
- ▶ Logon (give your username and password) and get the "ftp>" prompt. Because you are telneting to Humanitas as if

you were dialing in from a modem, you will need to pass the security hurdle of a secondary logon process. (When you dial in from home, the Telnet logon is your only logon; when you telnet to the server from a hardwired computer, Telnet requests the equivalent of a dial-in logon, though in this case it will be your second logon. This is because if Telnet did not always request clearance to use its transfer tools anyone could telnet in to the server and download anyone's files.)

Important Note: Your Telnet username and password on a hardwired machine (usually the same as those you use for Humanitas in other contexts) must previously have been set up in a special file in the C:\CUTCP directory on your faculty-office or department computer. By a quirk of the System Administrator's humor, the English Dept. convention is to call this file "bozo". Faculty members have individual bozo files; graduate students working on the communal machines in the department computing room add their logon information to a common bozo. (If you do not have a bozo or are not part of a common bozo, see Terri Jo Ortega to set things up.)

Once you are past the Telnet logon, you will be at the ftp> prompt indicating that the File Transfer Protocol (FTP) program is ready to go.

- Set FTP to "binary" mode if necessary. If the file you intend to download/upload is a text-only or ASCII file, proceed to the next step. (Note: word-processor documents are not ASCII files, but can be saved as such in most programs through an ASCII-save option that strips away formatting codes.) However, if it is a binary file (any program,

graphics, audio, compressed, or formatted word-processor file), first type "binary" to set FTP to binary transfer mode. FTP will stay in binary mode in the current session until you return it to text mode by typing "ascii" (in most circumstances, you can simply stay in binary mode once you have entered it because it works with text-only files as well).

- Download by typing "put [filename]", where "filename" is the name of the file you want to download from your Unix home directory. If the file is not in your home directory (on Unix directories, see §4.3–4.4), add a path name to the filename (on path names, see §4.4). FTP will report the result of the transfer and place the file under its original name in the C:/CUTCP directory on your local hard drive. The original file will still exist on Humanitas.

Note: most campus humanities computers have very small hard drives; make sure there is enough room on C: drive to store incoming files. Also, to avoid potential problems be sure that the file you are downloading has a name no longer than eight characters plus a three-character extension (i.e., the DOS "8.3" convention).

When downloading a Unix-file named, for example, "reflections_on_art", either change the name of the Unix file before you download (see §5.5) or download it to a new, shorter name. To download a file under a name different from the original, type "put [source-filename] [destination-filename]".

- Upload by typing "get [pathname\filename]", where "pathname\filename" is the file you want to upload from your local computer to Humanitas. The file can be on

any drive or directory on the local computer (e.g., you can "get d:\wp\letters\letter1"). FTP will report the result of the transfer and place the file on your home directory on Humanitas. To upload to a different directory or filename, go "get [source-pathname\filename] [destination-pathname/filename]". (Note that Unix uses a forward rather than backward "/" delimiter.) The original file will still exist on your local computer.

- Type "bye" at the ftp> prompt to exit. "bye" exits FTP and Telnet to return you to the humanitas> prompt.

🐼 The directionality of the terms "put" and "get" make sense only from the perspective of the server computer rather than user. A download is a "put" because from the server's viewpoint it is being asked to put something on a remote computer. And an upload is a "get" because to the server it appears that it is getting something from "out there."

2.3: FROM MICROCOMPUTER LABORATORY COMPUTERS

To download/upload from a Mac LC at the Microcomputer Lab, follow this procedure:

- Insert a floppy diskette in the drive.

- ▶ Open the "File" drop-down menu on the menu bar at the top of the NCSA Telnet 2.5 screen and make sure "FTP Enabled" and "Mac Binary Enabled" are selected.
- ▶ Open the "Network" menu and select the "SENDFTP" command. You will see something like the following:

```
ftp-n 128.111.158.10
Connected to 128.11.158.10.
220 Macintosh Residen FTP server, ready
```

- ▶ Open the "File" menu again and select "Set Transfer Directory". When the Set Transfer Directory dialogue box appears, hit the button marked "Desktop". In the next dialogue box that appears, your floppy drive should be listed as one of the options. Select it; then hit the button marked "Set Directory".
- ▶ At the mcl% prompt, type "ftp". The ftp> prompt will appear.
- ▶ To download a file from your storage space on the MCL server to your diskette, type "put [filename]". ("mput" downloads multiple files separated by spaces.) Use "put [source-filename] [destination-filename]" to download the file to a different filename.
- ▶ To upload a file from your diskette to your storage space on the MCL server, type "get [filename]". Use "get [source-filename] [destination-filename]" to upload the file to a different filename.

- Type "bye" at the ftp> prompt to return to the mcl% prompt.

2.4: FROM HOME PERSONAL COMPUTERS BY MODEM

To understand downloading/uploading by modem, it is useful to know just a little about the principle of serial transfer.

Think of Humanitas as a large reservoir of water connected to the smaller reservoir of a home computer by a slender pipe (the telephone line). Water, in this analogy, is data. A modem (which stands for "modulation/demodulation") is a device designed to funnel data through the pipe by means of "serial" transfer--i.e., one bit after another rather than many bits at a time as in "parallel" transfer. To download/upload files over a phone line (and to build in tolerance for line noise), modems use such transfer protocols as Kermit, Xmodem, Ymodem, and Zmodem. These protocols do the following: (a) break up the data stream into discrete "packets" that can be individually sent and, if necessary, resent; (b) add an extra amount of information to each packet for the purpose of "error-checking"; and (c) reassemble the packets at the other end.

There has to be a copy of the same transfer protocol at both sending and receiving ends for serial downloading/uploading to work. At present, the Humanitas server has only the Kermit protocol (managed by the program, "C-Kermit"), and is thus limited to transferring text-only or ASCII files by modem (for a workaround to the text-only limitation, see §6.7 on "uuencoding"). You can put a matching Kermit protocol on your home computer in two ways. One is to purchase a communications program that includes Kermit as

one of its internal protocols (most comm programs include the X, Y, and Zmodem protocols; check to be sure that the one you choose also includes Kermit). The other way is to acquire from a friend or from the Internet (though this would paradoxically already require you to be able to download) a freeware or shareware version of the communications program named "Kermit." This program exists in various machine-specific flavors (e.g., MS-Kermit for DOS computers). Anyone who does a lot of uploading/downloading should have both options: commercial programs are best for most uses, but their Kermit protocol tends to be slow (because not configurable); inversely, MS-Kermit and its kin have very primitive user interfaces but can be configured to match the Humanitas Kermit for best speed (the speed difference is significant).

The exact procedure for downloading/uploading via Kermit and a modem will vary depending on which communications program you are using at home. You should work your way manually through downloading and uploading with your program at least once so you understand the procedure, then set up scripts in that program to handle things automatically. For example, I have scripts in Bitcom and HyperAccess/5 (the two communications programs I have used) for downloading and uploading. Hitting Alt-R (for "receive") while running Bitcom on my home computer prompts me for the name of the Humanitas file to download and then automatically initiates the download. Alt-S (for "send") similarly uploads a file from my home computer to Humanitas. In HyperAccess/5, my <PageDown> and <PageUp> keys automate downloading and uploading, respectively.

Downloading:

The basic procedure for downloading is as follows:

- ▶ Type "kermit" while logged onto the server at the *humanitas*> prompt. This will start the server's Kermit and give you the "C-Kermit>" prompt.
- ▶ Type "send [filename]", where "filename" is the file you want to download from Humanitas. Add a pathname if the file is not in your present Unix working directory (by default, this is your home directory; see §4.4). You can also type "send [source-filename] [destination-path\filename]" if you want to download the file to a different directory or filename on your pc. (Normally, the file will be placed on your pc either in your current directory or, if you have set up your comm program and/or operating system this way, to a default directory you designate. Many comm programs put downloads into their own program directory unless you configure them differently.) Once you are finished entering the filename and hit <Return>, the C-Kermit program running on the server will tell you: "Return to your local Kermit and give a RECEIVE command. KERMIT READY TO SEND."
- ▶ Switch to your comm program's command-menu screen and select the file-transfer, Kermit-protocol, and receive-file options. Immediately after instructing the Kermit program running on the server to send a file, that is, pop up the command-menu screen for your comm program. (In Bitcom, for example, hitting <F2> during a communications session brings up the Bitcom menu.) Then select, in sequence, the file-transfer, Kermit-protocol, and receive-file options (or whatever

they are called in your program). (Note: you can simplify this routine in most comm programs by presetting file transfer to the Kermit-protocol. Also, the routine is not the same on all comm programs: in some, for example, you choose either receive-file or send-file from the main menu rather than first choosing the generic file-transfer option.) Once you initiate the receive-file option, your comm program will send a signal to C-Kermit on Humanitas saying that it can begin to send the file.

- ▶ After a pause, downloading should begin. Your comm program will display a screen showing the name of the file being downloaded, its size in kilobytes (K), the number of K transferred so far, the number of packets sent so far, the estimated time of transfer, time elapsed, etc.. At 2,400 baud, an average letter or memo takes a minute to transfer. An essay-length piece takes 4 to 5 minutes. Anything above 100 K in length takes a LONG TIME.
- ▶ "quit" from C-Kermit. When the download is complete, most comm programs alert you with a beep and ask for a <Return> or other key. Once you hit the key, the program will quit Kermit at its end, return to the communications session proper, and show you the C-Kermit> prompt of the server's Kermit program. Type "quit" to exit from the C-Kermit program to the humanitas> prompt.
- ▶ At your leisure, you can then retrieve the file (if it is a text file) into your word processor at home. Program, graphics, audio, and other binary files, however, often

require a further stage of processing because they often come in compressed form (with filenames that end, for example, "filename.zip"; see §6.7). It is also a good idea to run any downloaded program files through a virus checker program.

A typical download session looks like this (I elide the customized displays of your comm program):

<pre>humanitas> kermit C-Kermit 5A(189), 30 June 93, SunOS 4.1 (BSD) Type ? or HELP for help C-Kermit> send agrippa Return to your local Kermit and give a RECEIVE command. KERMIT READY TO SEND...</pre>	<p><i>[Here you call up the menu of your comm program, select the Kermit protocol if it has not been preselected, and initiate the receive-file option. The download will begin. Your comm program "observes" the transfer by displaying a status screen with file name, size, progress of the transfer, time elapsed, etc. When the download is done, the comm program will "beep" and ask that you hit some key. It will then return you to the C-Kermit program running on Humanitas.]</i></p>
<pre>C-Kermit>quit humanitas></pre>	

Uploading:

The basic procedure for uploading is as follows:

- ▶ Type "kermit" while logged onto the server at the `humanitas>` prompt. This will start the server's Kermit program and give you the "C-Kermit>" prompt.

- ▶ Type "receive" at the C-Kermit> prompt. C-Kermit will respond: "Return to your local Kermit and give a SEND command. KERMIT READY TO RECEIVE..."
- ▶ Switch to your comm program's command-menu screen and select the file-transfer, Kermit-protocol, and send-file options. Immediately after instructing the Kermit program running on the server to receive a file, that is, pop up the command-menu screen for your comm program. (In Bitcom, for example, hitting <F2> during a communications session brings up the Bitcom menu.) Then select, in sequence, the file-transfer, Kermit-protocol, and send-file options (or whatever they are called in your program). (Note: you can simplify this routine in most comm programs by presetting file transfer to the Kermit-protocol. Also, the routine is not the same on all comm programs: in some, for example, you choose either receive-file or send-file from the main menu rather than first choosing the generic file-transfer option.)
- ▶ Fill in a filename. Once you initiate the send-file option at your end, your comm program will ask you for a filename. If the file you want to upload is in the default directory of your comm program (so it knows where to look for it), "[filename]" is sufficient. Otherwise, supply a pathname (e.g., "wp51\letters\letter2"). Hit ^Return> when done entering the name.
- ▶ After a pause, uploading should begin. Your comm program will display a screen showing the name of the file being

uploaded, its size in kilobytes (K), the number of K transferred so far, the number of packets sent so far, the estimated time of transfer, time elapsed, etc.. At 2400 baud, an average letter or memo takes a minute to transfer. An essay-length piece takes 4 to 5 minutes. Anything above 100 K in length takes a LONG TIME.

- ▶ "quit" from C-Kermit. When the upload is complete, most comm programs alert you with a beep and ask for a <Return> or other key. Once you hit the key, the program will quit Kermit at its end, return to the communications session proper, and show you the C-Kermit> prompt of the server's Kermit program. Type "quit" to exit from the C-Kermit program to the humanitas> prompt.
- ▶ At your leisure, you can then retrieve the file (if it is a text file) into Pine e-mail through the "^R" option on the Compose Messages screen.

A typical upload session looks like this (I elide the customized displays of your comm program):

```
humanitas> kermit
C-Kermit 5A(189), 30 June 93, SunOS 4.1 (BSD)

Type ? or HELP for help

C-Kermit> receive
Return to your local Kermit and give a SEND command.

KERMIT READY TO RECEIVE...

[Here you call up the menu of your comm program, select the Kermit protocol if it has not been
preselected, and initiate the send-file option. Supply a filename and begin the upload. Your
comm program "observes" the transfer by displaying a status screen with file name, size, progress
```

of the transfer, time elapsed, etc. When the upload is done, the comm program will "beep" and ask that you hit some key. It will then return you to the C-Kermit program running on Humanitas.]

*C-Kermit> quit
humanitas>*

🐢 Studies have shown that Kermit, despite its reputation as a slowpoke compared to such transfer protocols as Zmodem, is actually the fastest for most purposes--provided that you are running it at your local end from the "Kermit" program itself (e.g., MS-Kermit for DOS, C-Kermit for OS/2) rather than as an internal protocol within a general purpose comm program, and provided also that you are willing to get under the hood and tinker a little. If you do a lot of downloading/uploading, it is well worth your while. The difference can be four or more times the speed. 🐢 To customize Kermit for best speed, use the pico text editor to create or edit the file named ".kermrc" on your home directory (see §6.2 on pico). Include the following lines in the file:

```
def bin set file type bin
def asc set file type text
set parity none
set flow rts/cts
set file byte 8
```

(Type "bin" before downloading a binary file; type "asc" to reset to ASCII transfer.)
Correspondingly, MS-Kermit for DOS on the home computer must be configured with a "mskermit.ini" file that includes the following statements under the File Transfer Settings section (consult the documentation or readme file for the equivalent settings in Kermit for Mac or OS/2):

```
set parity none
set flow rts/cts
```

```

set rec pac 9024
set send pac 9024
set buffers 28048 28048
set block 1
set window-size 2

```

On your home pc, edit the configuration or initiation file in your Kermit directory (in MS-Kermit 3.13, this file is called "mskermit.ini") so that it matches the above settings. Consult the documentation or readme file that came with your home Kermit for the appropriate syntax of the "set" commands. You should see a dramatic difference in speed. The key is the increase in "pac" or packet-size (so more data is sent in each "gulp" without pauses for error-checking). If you find that you are experiencing too many "resent" packets because of line noise (which makes large packets counter-productive), then try a smaller size. 🐼 For users with heavy on-line appetites, it is sensible to use both a general-purpose, commercial comm program and a shareware Kermit program--the former for most purposes, the latter just for long downloads/uploads.

CHAPTER 3:

GETTING OUT OF TROUBLE

3.1: HOW TO EXIT, ESCAPE, INTERRUPT

On the principle that the first step in driving the information highway is to learn where the exits are and how to ask for directions, here--even before learning about the Unix operating-system environment and the Internet--is an emergency kit.

There is no online experience more frustrating than finding oneself trapped in a dead-end process. Some common

examples: the logon routine gets stuck somewhere and leaves you in limbo; you've started to download a file that you suddenly realize is as big as Moby Dick; the nn newsreader freezes in the middle of retrieving a posting you want to read; you're stuck in a library catalog you've reached by anonymous telnet and there seems to be no way to exit; you've made a typing mistake and in this particular corner of the Internet the backspace key doesn't work; your phone line cuts out in the middle of an nn session and when you log back on and restart you receive an error message saying that "another nn process is already running."

When stuck in a circle of computing hell, try the following (in order of escalating seriousness):

- **^C** This is the universal (sort of) "cancel key." Pressing <Control> and "c" simultaneously will usually interrupt whatever is happening and return you to wherever you were when you started the current process. If that doesn't work, press "**^C**" a few more times.
- **^]** If **^C** doesn't work, try "**^]**". (In Telnet, this is the "escape" key to get back to the telnet> prompt.)
- **quit, exit, bye, stop, logoff, q** When stuck somewhere and the above measures don't work, or if you want to get out of a whole program (e.g., FTP, Telnet, Pine, etc.) rather than just a command in that program, take the more drastic step of trying these measures. Hopefully, one of them will dump you to the familiar *humanitas*> prompt.

- **<Break>**: If nothing else works, try the even more drastic "Break" key on your keyboard. It will dump everything, including the connection to Humanitas, and force you to logon again. (If you were connected by modem, the modem link to the campus terminal server should still survive and you can begin at the "connect hcfmail" step; see §2.4).
- **Worst Case Scenario** If worse comes to worse and you are connected by modem, invoke the command in your communications program that hangs up or disconnects the line. If you can't even get that to work, or if you are working on a computer hardwired to Humanitas, do a "hot reboot" of your computer (Control-Alt-Delete). And if even that fails, perform a "cold reboot" (turn off your computer and turn it on again). Usually, you'll be none the poorer after shutting down your connection (e.g., the e-mail message you were composing in Pine will usually still be there, waiting for you to complete it).

3.2: HOW TO "KILL"

Unix is an operating system that allows multiple processes to run simultaneously on "foreground" and "background" planes.

For most humanities users, there are few if any occasions to call upon the advanced procedures needed to start or push a process into the background; everything can usually be handled as if an online session had only one plane on which only one

process can happen at a time (e.g., a Unix command, e-mail, ftp, telnet, gopher, etc.).

However, there are circumstances in which a runaway or hung process necessitates disconnecting the online session and restarting from scratch (as per above). Sometimes, the runaway process that was the culprit will then appear to be continuing in the background, blocking any foreground use of that process. For example, when you interrupt a frozen nn session (without "quitting" the nn newsreader), logon again, and then try to restart nn, you may see an error message to the effect that "another nn process is already running".

If the problem doesn't clear up by itself in a few minutes (which it may or may not do), then you will have to "kill" it:

- First type **"ps"** at the *humanitas*> prompt to identify the processes currently running in your Unix session. You will see a response like the following:

PID	TT	STAT	TIME	COMMAND
16138	p6	S	0:00	-csh (csh)
16156	p6	R	2:31	nn
16165	p6	R	0:00	ps

In this example, the **ps** command reports that there are three processes running in the current Unix session: the Unix C-Shell (csh), the **ps** process itself, and nn. Make a note of the "process ID" number (PID) for the errant process you are interested in. In the above example, the PID for nn is "16156".

- Then type **"kill [PID]"**, where PID is the relevant process ID number. If this doesn't do the trick, then try for a so-called "sure kill" by typing **"kill -9 [PID]"**.

3.3: HOW TO ASK FOR ONLINE HELP

Though some parts of the Internet (and the Unix operating system that is the Net's primary environment) guide the user with on-screen command "menus," vast stretches of this electronic universe are as empty of visible clues as that lonely "humanitas>" prompt on the screen. If you are having trouble with a particular procedure or command, try the following online help resources:

- ▶ Most of the important Internet programs and interfaces (e.g., Telnet, Pine, Gopher, etc.) have custom help screens--usually accessed by typing "?" or "help" from within the program. Some have several layers of help ranging from simple command summaries to extensive, online manuals (as in the nn newsreader).
- ▶ For routine Unix commands (such as the commands that allow you to manage directories and files), use the online Unix manual by typing the following at the "humanitas>" prompt: "man [name of command]". For example, "man mv" brings up several screens of information regarding the syntax of the command for renaming/moving files.
- ▶ If you don't even know (or can't remember) the name of the command you are confused about, then try "apropos [keyword]". This is a shotgun command that hits everything even remotely resembling the target. It will turn up the name of every

command that either includes the keyword in its name or is otherwise relevant. Hopefully, you will recognize the command you need in the resulting melange and use the "man" command upon it.

PART II

SURVIVING UNIX

CHAPTER 4:

UNIX OVERVIEW

4.1: THE IDEA OF UNIX

The "humanitas>" you see after a successful logon means that you are sitting at the prompt of the Unix operating system that runs the Humanitas server computer as well as most other computers on the Internet. (More exactly, you are sitting at the prompt of the Unix "csh" or C-Shell, which is the default Unix command-interpreter on our system. There are other possible shells.) This means that you need some basic Unix sense and a handful of commands to navigate around, to manage the storage space assigned to you, and, in short, to survive. If you are familiar with DOS or some other operating-system on your personal computer, you will already have a "sixth sense" for the most essential Unix commands.

The most important, general thing to know about Unix is that contrary to its name it does not have one "personality" but many personalities. Schizophrenia, as it were, is its personality. Because it exists in many "flavors," was created piecemeal by thousands of programmers in decentralized fashion over many years and for many purposes, and is under the control of no centralizing commercial company, Unix (together with its complement of common utility and client programs) is an operating system that often lets you accomplish any specific task in two, three, four, or more subtly incommensurable ways, each of which has multiple sub-options. Flexibility--as well as maddening diversity and inconsistency--is the name of this particular game. Unix is the great example of what Deleuze and Guattari call "multiplicity" or the "body without organs." Some examples of Unix schizophrenia: "w," "who," and "finger" all give you a list of users currently logged onto Humanitas, though

in different formats and with slightly different information. If you want to see the contents of a text file on your screen, you can use the "**more**," "**pg**," or "**cat**" commands, each of which behaves differently. There are umpteen ways to list (**ls**) your files and directories. And you can use either a numeric or alphabetic designator with **chmod** to change the permissions assigned to a file or directory.

Nevertheless, there is a core of Unix-conventions that you can pick up by osmosis. Eventually, you can gain a sort of "Unix sense" enabling you to make good guesses about what to do in any particular situation. In any case, don't be afraid to try things. It is seldom the case that Unix (in its default mode) will allow you to issue a command for an irrevocably destructive act without prompting for a "yes/no" confirmation.

4.2: YOUR ASSIGNED STORAGE SPACE

Each Humanitas user is assigned 1 megabyte (1 Mb or 1,024K [kilobytes]) of storage space on the Humanitas server (this translates to 2,000 "blocks" in the unit of measure reported by Unix; a block is a half-K or 512 bytes). In addition, no more than 200 files and/or directories can be kept in that assigned space. The file-number quota is not an issue (there are no reasons I can think of why you would ever need to keep this many files active in your account), but the storage-space quota is. It amounts to the fact that you have a very finite amount of physical harddrive in which to store both necessary system files and any e-mail folders (each folder, technically, is a file), data files, or program files you have acquired. By way of comparison, a standard article-length essay of 30 pages in a word-processor file is roughly 50-70K. A standard .gif picture is anywhere from 125-400K. And freeware or shareware programs (often downloaded over the Internet) can bulk as large as 400K or more. If you exceed your assigned space during any process (such as downloading a file), you will first see a warning on your screen. At this point, you are allowed a certain extra space beyond your quota (up to what the system calls your "limit") to finish your current process. However, if your process then goes

☹️ There are three commands you can use at the humanitas> prompt to monitor your storage space. (1) The **quota -v** command displays your current used space, assigned quota, and "limit." If you are over quota, the **quota** command by itself (without the "-v" switch) returns a message like the following: "Over disk quota on /home, remove 107K within 7.9 days." (2) To see an accounting of your used space broken down by directories, type **du** while in your home directory. (3) The **ls -a -s** command lists all files and subdirectories in your current working directory with their respective sizes (however, the size figure for subdirectories is not meaningful because it represents only the bulk of the directory apparatus itself--as if measuring the weight of an envelope but not of the materials inside).

on to exceed the limit it will be stopped midway with a message of the sort, "Limit exceeded -- write failed." This means you need to delete some of your files and restart your process from scratch.

For some purposes, you may want to take advantage of a public access directory on Humanitas called "/tmp" (the MCL server equivalent is `/temp`). For example, you can download a huge file via FTP from a remote server to `/tmp`, then download the file in a second stage from `/tmp` to your home computer. The advantage is that you can download files too big for your personal storage space to handle; the disadvantage is that `/tmp` is a temporary directory whose contents are periodically purged by the system.

4.3: DIRECTORIES AND FILES

As in the case of other operating systems, storage space in Unix is organized by "directories" and "files." A directory is a container structure in which you keep files and/or further subdirectories. Each subdirectory can then itself include its own files and sub-subdirectories and so on in a regression that is for practical purposes infinite.*

Files are of two basic kinds: text files and binary files. Text files consist of "ASCII" characters limited to the alphanumeric set and the non-alphanumeric dingbats you might find on a typewriter (plus a few, basic line-drawing and box-drawing characters). Text files can contain, for example, the basic substance of a word-processing document without any formatting codes. They can be viewed in Unix using the "`more`" or "`pg`" commands (see §6.1). Binary files, which appear as gibberish when viewed

* For convenience, this manual usually refers to directories or subdirectories at whatever level of subordination simply as generic "directories."

through "more," "pg," or some other text viewer, are usually executable program files or graphics files.

☛ Unix allows you to create directories and files with names up to 14 characters long (actually, you can create longer names, but only the first 14 characters are functional). In practice, you should usually restrict file names to no more than 8 characters (plus an optional extension consisting of a period and 3 characters) to prevent problems when downloading to DOS. DOS recognizes only an "8.3" naming format. You can use any character in a name except a space or the forward slash

4.4: TREE STRUCTURE

As in other operating systems, the overall organization of directories and files on the physical storage space of the Humanitas server is known metaphorically as a "tree." There is a "root" directory for the whole tree and then branch directories that ramify from the root (as well as sub-branches that ramify from the branches, and so on). Visualized a different way, subdirectories do not so much depend from their parent directories as nestle "within" them. Thus a directory is said to "contain" subdirectories, which in turn can contain further subdirectories.

A "path" is the prefix to any directory or file name that maps the route from the root through all intervening directories to that directory or file. The root directory in Unix is represented simply by "/". On the Humanitas system, the root contains, among other things, a directory whose path is "/home" (meaning "the directory named 'home' that branches from the "/" directory).

And within "/home" are all the particular "home directories" of Humanitas users. For example, the path for my personal home directory is "/home/ayliu" (meaning: the directory named "ayliu" that branches from the "home" directory that branches from "/").* An individual user controls his/her home directory and any of its included subdirectories; users can access other parts of the tree only with the permission of those who control those spaces (see §7.1 on permissions). The full path name of any directory or file is known as an "absolute path." For convenience, Unix also allows the use of "relative paths." If you are presently in the /home/ayliu directory, for example, you can simply type "cd backups" to change your working directory to /home/ayliu/backups/. It is understood that you are using the "cd" or change directory command from the "perspective" of /home/ayliu. (On the cd command, see below.)

When you logon and get the humanitas> prompt, your initial working directory--or, to borrow a film concept, point-of-view (pov)--is by default

☛ DOS users should notice that Unix uses the forward-slash ("/") for the directory delimiter while DOS uses the backward-slash ("\"). Why? As that movie about Australian "savages" says, "the gods must be crazy." Also note that the "ls" command (see below) lists directory names with a trailing slash to prevent confusion with file names (thus "News/" is the subdirectory in my home directory whose full path is

* Note: the term "home directory" conventionally means the "/home/[username]" directory and not the "/home" directory above it.

your home directory: `"/home/[username]"`. Any directory or file management commands you issue will assume--unless you specify differently--that you mean the action to be undertaken in the context of your home directory. Thus if I am in my home directory, any file I create under the name `"[filename]"` will be stored in `/home/ayliu` as `"/home/ayliu/[filename]"`. To store it elsewhere, I would have to add a path to the filename (e.g., `"/home/ayliu/backups/[filename]"`.) As described below, you can easily "change directories" or move your point-of-view. In this case, your commands will be enacted in the context of your new location.

Most of the time, you will be operating in your home directory. Occasionally, you will need to create or move to a subdirectory. And even less frequently, you will need to move to someone else's home directory or elsewhere in the system to read or copy a file (e.g., to the `"/home/ayliu/for_all_hands"` directory in my own Humanitas storage space where I keep certain program and configuration files useful to all).

CHAPTER 5:

DIRECTORY MANAGEMENT COMMANDS

5.1: WHERE ARE YOU? (pwd)

To locate where you are in the tree at any time (your current working directory), type "**pwd**" at the Humanitas prompt. This command, which means "print working directory," will return your location. (For reasons having to do with the state of hardware when Unix originated, "print" often means "print to screen" or "display.")

For example:

```
humanitas> pwd
/home/ayliu
```

🐼 Not just directory or file names but also commands are case-sensitive in Unix. Thus "**pwd**" is the right command; "**PWD**" doesn't work

5.2: LISTING THE CONTENTS OF A DIRECTORY (ls)

To list the contents of your current working directory, use the "**ls**" family of commands. The most important are the following:

- **ls**

Typing "ls" at the *humanitas*> prompt lists the directories and files in your current working directory. For example, my home directory presently holds the following subdirectories and files:

```
humanitas> ls
News/      d2          for_all_hands/  mail/
agrippa for_all_eyes/  info           web
```

► ls -a

Typing "ls -a" ("ls" with the "-a" switch) means "list all" directories and files in your working directory. In addition to normal files and directories, this command shows "system files" (preceded by a dot: "."). System files are used to set default configurations and options for various processes. Some system files are necessary for every user. For example ".cshrc" sets the default options for the Unix shell. Other system files are automatically created by a particular process when it is run for the first time. For example, running the newsreader program on Humanitas (nn) sets up a ".newsrc" file that keeps track of Usenet groups. And still other system files are created at the user's discretion. For example, you can create a ".signature" file that adds an automatic signature at the bottom of your e-mail letters. As shown by the "ls -a" command, my home directory presently looks like this:

```

humanitas> ls -a
./          .gopherrc~  .newsrc.bak  .plan      for_all_hands/
../         .history   .nn/         .project   info
.addressbook .kermrc    .pine-debug1 .signature  mail/
.backups/    .login     .pine-debug2 News/       web
.cshrc       .logout    .pine-debug3 agrippa
.exrc        .mailrc    .pine-debug4 d2
.gopherrc    .newsrc    .pinerc     for_all_eyes/

```

► `ls -l`

Typing "`ls -l`" ("`ls`" with the "`-l`" switch) means "list in long format." This command shows you a list of your directory contents as in the following example:

```

humanitas> ls -l
total 32
drwx----- 2 ayliu      512      May 24 00:22  News/
-rw----- 1 ayliu     13073     Jan  2  1993  agrippa
-rw----- 1 ayliu     1919     May 25 21:07  d2
drwxr-x--- 2 ayliu      512      Sep  9  1993  for_all_eyes/
drwxrwx--- 2 ayliu      512      May 22 18:14  for_all_hands/
-rw----- 1 ayliu     3372     Jan 28 16:38  info
drwx----- 2 ayliu      512      May 24 21:27  mail/
-rw----- 1 ayliu     8729     Feb  9 23:13  web

```

Here, "Total 32" means that a total of 32 blocks (16K) are used by the contents of the directory (not counting the contents of any subdirectories). Each of the succeeding lines then contains this information:

Example

Meaning

`-rw-----`

These are the permissions assigned to a directory or file (a "d" at the start means the entry is a directory rather than file).

"-rw-----," for example, means that this is a file and it can be read and written to by the user but by no one else. (See §7.1–7.2 on permissions.)

2 ayliu In the case of directories, the number indicates how many files the directory contains. The letters are your username.

13073 This number reports the size of the file or directory in bytes (13073 bytes=about 13K). In the case of directories, the number reports only the space used by the actual apparatus of the directory and not by its contents.

ayliu This is your username (also known as "userid").

May 24 00:22 This is the date and time the file was created or last revised. (Unix works on a 24-hr. clock).

agrippa This is the name of the file or directory.

► ls -R

Typing "`ls -R`" ("`ls`" with the "`-R`" switch) lists files and directories "recursively." This means that if you are in your home directory, you will see a list of that directory's files and

subdirectories followed by separate contents lists for each of the subdirectories (and so on into any sub-subdirectories, etc.).

💡 Most Unix commands have optional "switches." Usually, these switches can be combined. Thus, for example, "`ls -l -a`" lists in long format all the files/directories in the current working directory (including any system files).

5.3 CREATING OR MAKING DIRECTORIES (mkdir)

To create a directory, use the make directory command: "`mkdir [directory]`." For example, the following creates the directory `/home/ayliu/backups/`.

```
humanitas> mkdir /home/ayliu/backups
humanitas>
```

After you have entered your command, Unix returns the `humanitas>` prompt again to indicate that you have succeeded.

Note that the above example makes use of an absolute or full path name. For convenience, Unix also allows relative paths. If you are already in the `/home/ayliu` directory, for example, "`mkdir backups`" creates `/home/ayliu/backups`.

When making a directory, all parent directories of the final, intended subdirectory branches must already exist. Thus

to create `/home/ayliu/backups/july/dos_programs`, each of the directories `/home`, `/home/ayliu`, `/home/ayliu/backups`, and `/home/ayliu/backups/july` must already have been created. As a shortcut, you can use `"mkdir -p [path/directory]"`, which automatically creates all intervening directories in the path if they don't already exist.

5.4: CHANGING DIRECTORIES (`cd`)

To change your current working directory, use the change directory command: `"cd [target directory]"` (`"cd"` is short for `"chdir"`). For example, the following moves me to my News subdirectory:

```
humanitas> cd /home/ayliu/News
humanitas>
```

Again, Unix allows relative path names. If you are already in `/home/ayliu`, `"cd News"` moves easily to `/home/ayliu/News`. Also for convenience, the `"cd"` command by itself returns you from wherever you happen to be to your home directory.

To confirm that you are where you want to be, use the `"pwd"` command.

5.5: RENAMING DIRECTORIES (`mv`)

To rename a directory, use the move command: `"mv [original directory name] [new directory name]"`. For example, the following renames my News subdirectory to `"badnews"`:

```
humanitas> mv /home/ayliu/News /home/ayliu/badnews
```

```
humanitas>
```

Note: the "move" command is so-named because it can also be used to move files simply by renaming them with a different path (see "File Management Commands" below). However, **mv** as applied to directories allows only renaming and not moving (that is, you can rename a directory so long as the new name occupies an equivalent level in the tree, but you cannot rename it so that it jumps to a different level in the tree; renaming the `/home/ayliu/News` directory to `/home/ayliu/badnews` is allowed, but not renaming it `/home/ayliu/under_the_rug/badnews`). This is because the consequences of moving a whole directory and its contents from one level in the tree to another are so much graver. There is a separate **mvdir** command that allows the moving of directories together with all their included subdirectories, but there are usually no practical reasons to use it.

5.6: REMOVING OR DELETING Directories (**rmdir**)

To delete a directory, use the "remove directory" command: **rmdir [directory]**. For example, if I am in my home directory, the following deletes the directory named `/home/ayliu/temporary` using the relative path name:

```
humanitas> rmdir temporary
```

```
humanitas>
```

For safety, any directory you remove using the **rmdir** command must be empty of files or subdirectories. Otherwise you will

receive a message on the screen saying it isn't, and the removal won't be performed. You can either first delete the contents of the directory and then delete the directory itself or use the command `rm -r [directory]`. This forces the deletion of the directory and all its contents. Do not use this command unless

🔒 In general, be careful with any Unix command that involves removing or deleting. There is no "undo" feature in Unix. Once you delete a directory or file, there is absolutely no way to get it back. It is a good idea to create a backup directory and copy any crucial file to it before you edit or otherwise tinker with it.


you are absolutely sure you mean to do so.

CHAPTER 6:

FILE MANAGEMENT COMMANDS

6.1: VIEWING CONTENTS OF EXISTING FILES (more or pg)

Existing files can be displayed on the screen a page at a time with the "`more [filename]`" or "`pg [filename]`" commands. The major difference is that `more` is a one-way street: you can only read forward in the file a screen at a time by pressing the spacebar. The `pg` command is a much more useful browser, allowing you to move both forward and backward a screen at a time; `<Return>` moves forward, and the hyphen or minus key ("`-`") followed by `<Return>` moves backward. `pg` also has other handy features. For example, "`+[n]`" and "`-[n]`" move ahead and back "*n*" number of pages, respectively (e.g., "`+3`" moves forward 3 pages). There is also a search function: "`/[text-string]`" jumps to the next appearance of the specified text-string (e.g., "`/access`" searches for "access"). Although some interfaces to the Internet (e.g., Gopher) are set by default to use

 You may occasionally want to use the `head [filename]` and `tail [filename]` commands, which display just the first and last ten lines of a file, respectively. The `cat [filename]` command can also be used to scroll a file across the screen, but since `cat` does not pause in scrolling it is useful only for

more whenever they need to display a text file, they can often be reset to use *pg* if you wish.

Hitting "*^C*" (<Control> and *C*) at any time interrupts either *more* or *pg* and returns you to the *humanitas*> prompt.

Here is an example of *more* in action. The first screen of the file "sterling" in my home directory (an article by Bruce Sterling about the Internet) is displayed with a line at the bottom indicating roughly how much of the file has been displayed; pressing the spacebar advances to the next screen of the file:

```
humanitas> more sterling
Short History of the Internet by Bruce Sterling

bruces@well.sf.ca.us

Literary Freeware -- Not for Commercial Use

From THE MAGAZINE OF FANTASY AND SCIENCE FICTION, February 1993. F&SF, Box 56,
Cornwall CT 06753 $26/yr USA $31/yr other

F&SF Science Column #5 "Internet"

Some thirty years ago, the RAND Corporation, America's foremost Cold War think-tank, faced a
strange strategic problem. How could the US authorities successfully communicate after a
nuclear war?

Postnuclear America would need a command-and-control network, linked from city to city,
state to state, base to base. But no matter how thoroughly that network was armored or
protected, its switches and wiring would always be vulnerable to the impact of atomic bombs. A
nuclear attack would reduce any conceivable network to tatters.

And how would the network itself be commanded and controlled? Any central authority, any
network central citadel, would be an obvious and immediate target for an enemy missile. The
center of the network would be the very first place to go. RAND mulled over this grim puzzle in
deep military secrecy, and arrived at a daring solution. The RAND proposal (the brainchild of
RAND staffer Paul Baran) was made public in 1964. In the first place, the network would
*have no central authority.* Furthermore, it would be *designed from the beginning to operate
while in tatters.*
```

The principles were simple. The network itself would be assumed to be unreliable at all times. It would be designed from the get-go to transcend its own
 --More--(5%)

And here is an example of *pg* in action. *pg* puts a colon (":") at the bottom of each screen, indicating that it is waiting for you to press <Return>, "-", "+[n]", "-[n]", or "/[text-string]".

For the sake of interest, I use *pg* below to move to the second screen of the file "sterling" in my home directory (the first screen is elided to save space):

```
humanitas> more sterling
```

```
[First screen of file elided here.]
```

```
:<Return>
```

unreliability. All the nodes in the network would be equal in status to all other nodes, each node with its own authority to originate, pass, and receive messages. The messages themselves would be divided into packets, each packet separately addressed. Each packet would begin at some specified source node, and end at some other specified destination node. Each packet would wind its way through the network on an individual basis.

The particular route that the packet took would be unimportant. Only final results would count. Basically, the packet would be tossed like a hot potato from node to node to node, more or less in the direction of its destination, until it ended up in the proper place. If big pieces of the network had been blown away, that simply wouldn't matter; the packets would still stay airborne, lateralled wildly across the field by whatever nodes happened to survive. This rather haphazard delivery system might be "inefficient" in the usual sense (especially compared to, say, the telephone system) -- but it would be extremely rugged.

During the 60s, this intriguing concept of a decentralized, blastproof, packet-switching network was kicked around by RAND, MIT and UCLA. The National Physical Laboratory in Great Britain set up the first test network on these principles in 1968. Shortly afterward, the Pentagon's Advanced Research Projects Agency decided to fund a larger, more ambitious project in the USA.


The nodes of the network were to be high-speed supercomputers (or what passed for supercomputers at the time). These were rare and valuable machines which were in real need of good solid networking, for the sake of national research-and-development projects.

In fall 1969, the first such node was installed in UCLA. By December 1969, there were four nodes on the infant network, which was named ARPANET, after its Pentagon sponsor. The four computers could transfer data on dedicated high-speed transmission lines. They could even be programmed remotely from the other nodes. Thanks to ARPANET, scientists and researchers could share one another's computer facilities by long-distance. This was a very handy service, for computer-time was precious in the early '70s. In 1971 there were fifteen nodes in ARPANET; by 1972, thirty-seven nodes. And it was good.

```
:
```

Note that **more** and **pg** can only meaningfully display text files--i.e., ASCII files containing the normal alphanumeric characters plus a few dingbats. When used on a binary file (any executable program, graphics or sound file, compressed file, or uuencoded file) **more** and **pg** display what appear to be a garbage characters--e.g.:

```
"%*djUFncI^_-AkjdptjbITNld-jl(?>_6djfUt5fUn)8*7d&&]%%$".
```

 In addition to displaying files stored on the Humanitas server, **more** and **pg** can be used with the "pipe" function ("|") to display files on remote computers. For example, while connected via "anonymous FTP" to a FTP file server, you can change to the remote directory you are interested in and go "get [filename] |pg". This means, "get the contents of [filename] from the remote computer and pipe them to the pg program on Humanitas which will display them on screen".

6.2: CREATING AND EDITING FILES (using Pico)

Ordinarily, users create ("write," "save") files in their Unix space as part of a larger process of transmitting files from remote computers through the Humanitas machine to their personal computer ("downloading") or vice versa ("uploading"). Three common examples: 1. you use "anonymous FTP," Gopher, Usenet or some other component of the Internet to write a program, picture, or text document from a remote

machine to your space in Humanitas, then--in a second step--you download the file from Humanitas to the hard drive on your personal computer. 2. You save an e-mail message someone sent you as a separate Unix file on Humanitas, then download it to the hard drive on your personal computer. 3. You create a text document in your favorite word processor, then upload it from your personal computer to the Humanitas machine for inclusion in an e-mail message. All these processes require that your Unix space on Humanitas serve as a sort of holding pen for files (which can then be kept or erased at your discretion). (Downloading and uploading are covered in §2.1-2.4.)

In addition, users can create simple text files from scratch using a variety of Unix text editors. The classic Unix editor program is "vi." A less powerful but much friendlier editor is "pico," which is also the default editor used to compose e-mail messages in Pine. One of the most common uses of such editors is to create (or edit) the system or "dot" files necessary to set

☛ Such processes as Pine (the most common e-mail interface) or nn (the Usenet newsreader on Humanitas) create certain Unix files automatically in the normal course of operations. Pine, for example, saves recently sent mail in a file named "sent-mail" in each user's mail/ directory; and nn creates a file named .newsrsrc to keep track of a user's subscribed newsgroups. In normal usage, such process-created files are managed through the interface of the particular process itself.

options for e-mail, Usenet, kermi, and other processes. For example, you can use a Unix text editor to create a .signature file to be automatically appended to your e-mail letters. Any other short or simple editing job is a candidate for the Unix text editors, which by comparison with current word processors are starkly primitive and not suitable for complex work.

Only Pico is covered here. The `pico` command by itself (without any parameters) calls up a blank screen with a menu of commands at the bottom. Write and edit in the blank space with the aid of these commands and the cursor movement keys (arrow up, down, left, right). Particularly useful editing commands include `^k`, which deletes or "kills" the line at the cursor and `^u`, which restores the last-killed line (or consecutive series of last-killed lines). Also, you can use `^k` and `^u` in concert with the block-begin command, `^^` (i.e., `<Control>^`). Move the cursor to the desired location and go `<Control>^` to mark the beginning of a block; then move the cursor to the end of the desired block (the material will be highlighted on screen). Once the block has been marked, you can delete it with `^k` or move it by going `^k`, moving the cursor to a new location, and using `^u` to restore the deleted material.

In the following example, I invoke Pico and use a blend of asterisks and alphanumerics to create an e-mail signature:

```

humanitas> pico
UW PICO(tm) 2.3                      New Buffer

*****

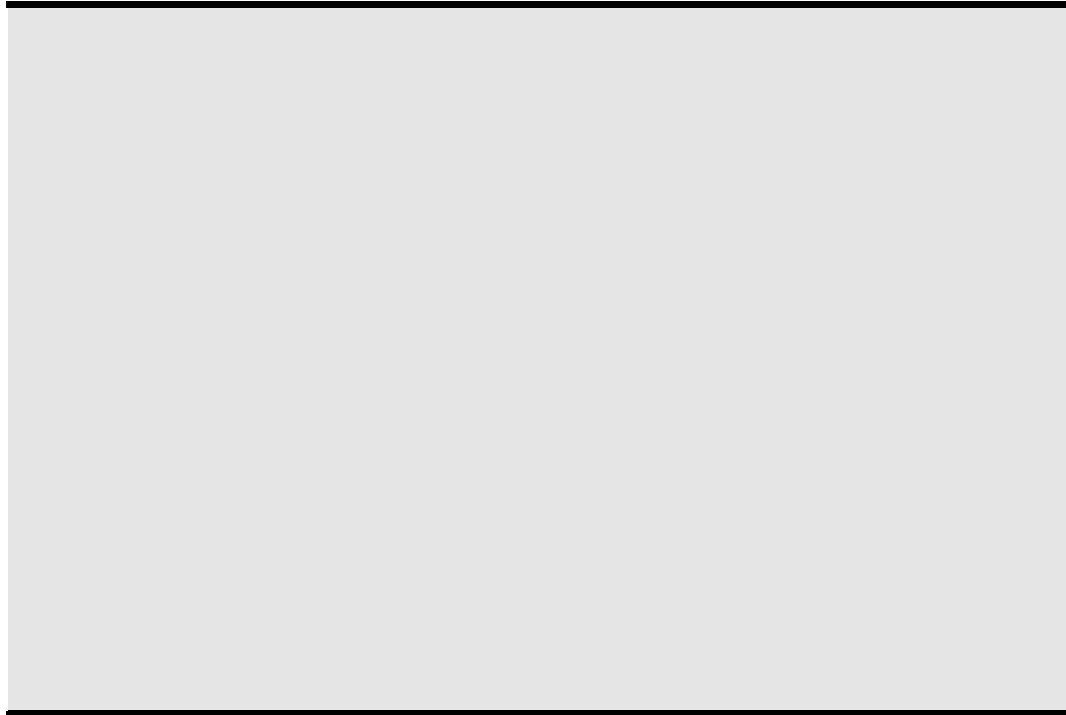
ALAN LIU                               ayliu@humanitas.ucsb.edu
Dept. of English                       FAX: 805/893-4622
Univ. of California
Santa Barbara, CA 93106
*****

^G Get Help   ^O WriteOut   ^R Read File   ^Y Prev Pg   ^K Cut Text   ^C Cur Pos   ^X
Exit          ^J Justify    ^W Where is   ^V Next Pg   ^U UnCut Text ^T To Spell

```

Editing existing text files is just as easy. Instead of using the *pico* command by itself, type *pico [filename]*. This will bring up Pico with the specified file loaded.

When finished creating or editing a file, use the *^O* command (<Control>O) to save the file. Pico will prompt for a file name and save to the current working directory if no other path is included. *^X* then exits from pico to the Humanitas prompt. Note: *^X* exits cleanly only if the file you are working on has not been modified in the current pico session. If the file has changed, *^X* first asks if you want to save the file and, if so, under what name.



6.3: RENAMING AND MOVING FILES (mv)

Because renaming and moving are conceptually the same operation in Unix, they share the same "move" command: `mv [original path/original filename] [new path/new filename]`.

To rename a file without moving it, simply designate a new filename with a path identical to the original file. If the file you are renaming is in your current working directory, then you need not specify any paths at all.

For example, the following renames the file "agrippa" in my current working directory to "cyberpoetry":

```
humanitas> mv agrippa cyberpoetry
humanitas>
```

To move a file to somewhere else in the tree, simply include a new path. If I am currently in my home directory, for example, the following moves the file "cyberpoetry" to my `for_all_hands` directory:

```
humanitas> mv cyberpoetry for_all_hands/cyberpoetry
humanitas>
```

Even simpler, since the new file will have a different path but the same filename as the original, you need only specify the new directory:

```
humanitas> mv cyberpoetry for_all_hands/
humanitas>
```

Note: To facilitate moving multiple files to a different directory, **mv** allows you to designate more than one file name on the command line (each separated by a space). The following moves three files named "d1," "d2," and "d3" from the current working directory to the "backups" directory:

```
humanitas> mv d1 d2 d3 backups/
humanitas>
```

🐼 For convenience, Unix allows the use of the following wildcard characters in file names: ***** (meaning any number of characters), **?** (meaning any one character), **[xyz]** (meaning any character from among the specified set, in this case "x" "y" and "z"), and **[x-z]** (meaning any character in the range specified, in this case from "x" to "z"). Thus "**mv d* for_all_eyes/**" moves all files that start with "d" to the "for_all_eyes" directory. The command "**mv d*.bak for_all_eyes/**" does the same for files beginning with "d" and

6.4: COPYING FILES (cp)

To copy a file, use the command: `cp [original filename] [new filename]`. When copying a file to a different directory, add path information: `cp [original path/original filename] [new path/ new filename]`.

For example, the following copies the file named "Baudrillard" in my current working directory to a file named "Sim_Baudrillard" in the same directory:

```
humanitas> cp Baudrillard Sim_Baudrillard
humanitas>
```

Using path information, the following copies "Baudrillard" to "Sim_Baudrillard" on a different directory (my "backups" directory):

```
humanitas> cp Baudrillard backups/Sim_Baudrillard
humanitas>
```

When copying files to a different directory, `cp` allows you to designate multiple files (each separated by a space). For example, the following copies three files named "d1," "d2," and "d3" from the current working directory to the "backups" directory:

```
humanitas> cp d1 d2 d3 backups/
humanitas>
```

(Note: the `copy` command accomplishes the same tasks as `cp` and, in addition, allows you to copy whole directories.)

6.5: CONCATENATING FILES (cat)

The "catenate" or "concatenate" command has a number of different uses, one of the most common of which is to combine multiple files into a single file as follows: `cat [file1] [file2] [file_etc.] > [new filename]`. The files you designate are "chained" together such that the end of "file1" is followed by the beginning of "file2," etc.

For example, let's say you have saved several e-mail messages received from John as separate Unix files titled "john1," "john2," and "john3" (using the "export" feature in Pine; see §9.3–9.4). You could then combine the files into a single file named "john_letters" as follows:

```
humanitas> cat john1 john2 john3 > john_letters
humanitas>
```

Sometimes you will need to edit the resulting file to adjust the way the original files were joined (e.g., you may want to add or remove line breaks between the files, delete headers from e-mail messages, etc.). To do so, just open up the file using Pico or another Unix text editor.

6.6: DELETING OR REMOVING FILES (rm)

To delete a file, use the "remove" command: `rm [filename]`. For safety's sake, `rm` is set by default (through the "alias rm = 'rm -i' " line in your `.cshrc` file) to prompt for confirmation before actually deleting the designated file. For example, the following deletes the file named "trial" in the current working directory:

```
humanitas> rm trial
rm: remove trial? y
humanitas>
```

To delete multiple files, list them one after the other separated by spaces or use a wildcard. Thus "`rm trial1 trial2 trial3 [etc.]`" and "`rm trial*`" both delete all files whose name begins with "trial" in the current working directory. Unix will prompt for confirmation file by file.

6.7: UUENCODED, COMPRESSED, AND ARCHIVED FILES

There are several, special kinds of files you will run into on the Internet. The most frequent are uuencoded, compressed (usually "zipped") and archived (usually "tar"red) files.

Uuencoding

Uuencoding is a work-around tactic for dealing with the problem that most users have one or more weak links in their communications setup where text-only files can cross but binary files (executable

programs, formatted text, graphics, sound files, etc.) are stopped. You can send ASCII text to someone by

e-mail, for example, but (except via another work-around called a "MIME" attachment; see §9.4) not a formatted word-processing document or program. Similarly, you can

👤 For late-breaking news regarding the downloading/uploading of binary files, see the box at the end of §2.4.

currently download ASCII text from the Humanitas server to your home computer via modem and the Kermit protocol, but nothing else. (Note: this may change in the future.)

The solution to this impasse is a very clumsy and indirect trick whose sole virtue is that it works. Binary files can be encrypted by "uencoding" so that text-only characters (e.g., "JdjUJgfui*&jd^_[dj!@_jdnbhUyrj&#") carry binary information.

Uencoded files can then be e-mailed or downloaded by modem and "decoded" on the other side to regenerate the original binary file.

To take advantage of uencoding, you need to use both the Unix uencoding program on the Humanitas server and an equivalent uencoding program on your local computer (e.g., "uuxfer" for DOS or "uuparser" and "uutool" for Mac). If you are working on an office or department computer hardwired to Humanitas, you can locate a uencoder program on the Internet (see §11.6), FTP it to a file on Humanitas (see §11.1-11.4), and then download it to a diskette in binary code (see §2.2). If you can only access the Internet via modem, however, you cannot download a uencoder (which is a binary file) without a uencoder (catch-22!); thus you will need to get it from some friend on a physical diskette.

Case 1: Let's say you are exploring the Internet through anonymous FTP, Gopher, Usenet, or some other interface (discussed in Part III) and come upon a program or picture file you would like to bring home. You first transfer the file from wherever it is on the global network and save it to a Unix file in your Humanitas storage space as "binary_file" (using the commands appropriate to the particular interface). Next, at the `humanitas>` prompt type `"uencode binary_file binary_file > binary_file.ue"`. This uencodes the binary file under the "binary_file.ue" filename (when uencoding zip compressed files,

go "uuencode binary_file.zip binary_file.zip > binary_file.uue"). Then download the resulting "binary_file.uue" to your local or home computer. Finally, call up the DOS or Mac uuencoding program on your local or home computer to decode the file and restore it to binary format. (For example, uuxfer for DOS displays a menu on which the "d" option prompts for a file to decode; the uuencoded file will be restored to its original, binary form under its original name.)

Case 2: Let's say you want to upload a formatted word-processor document named "binary_file" from your local or home computer to Humanitas for dissemination. First use the DOS or Mac uuencoding program on your local or home computer to uuencode the file. (For example, uuxfer for DOS displays a menu on which the "e" option asks for a file to encode; a file named "binary_file" will be encoded to the filename "binary_file.uue".) Then upload the resulting .uue file to the Humanitas server. From that point on, the file can be sent by e-mail or copied by other users (provided, of course, that the other party has a uuencoding program to decode the file at their end).

Compressed Files

To save on storage space and downloading time, many files on the Internet (usually program files) exist in compressed form. The most common compression format is "zip" (any file named "file.zip" is in compressed format). Zip and other compressed files are binary files that must be downloaded to your local or home computer using the uuencoding procedures described above. Once reproduced and decoded on your home or local computer, however, they will still exist in zipped form, which is unusable unless you "unzip" them. To do so, you will

need a pkzip/pkunzip program (or equivalent) on your pc. These can be downloaded from the Internet (you will need uuencoding to get them). You may also want to scout your existing personal computer software for copies of pkzip (many commercial software products now come in compressed form and include a copy of pkzip or its equivalent to load the program to your hard drive).

Once you have an unzipping program, then you can go "unzip [filename.zip]" to uncompress a zipped file. (Caution: it is usually best to isolate a zip file in its own directory before unzipping. This is because many zip files uncompress not into a single file but a whole constellation of files; these can become lost in an already populated directory and/or overwrite existing files with the same name.)

Archived Files:

Unix includes a utility named "tar" for archiving whole directories of files into a single file (useful for packaging, for example, an entire complex program with all its necessary subdirectories). The result is a binary file called "filename.tar". Sometimes, you will even see something like "filename.tar.zip", indicating that the material has been both tar archived and zip compressed. The complexities of tar cannot be covered here. The basic procedure for unpacking tar files is as follows. First, retrieve the file from wherever it is on the Internet to your Humanitas storage space (e.g., using anonymous FTP in binary transfer mode). Then, at the humanitas> prompt, type "tar -xvf [filename]".

CHAPTER 7:

DIRECTORY AND FILE PERMISSIONS

7.1: "OWNERSHIP" IN UNIX

So strong is the concept of property that even before the economics of the Internet have been worked out "property" is at work as metaphor. Directories and files in Unix are spoken of as "owned" by the users they are assigned to. Every user has ownership rights to his or her home directory as well as all included subdirectories and files. In addition, there is one superuser--the system administrator--who has right of eminent domain over everyone's property.

Ownership in this context means that the user has the ability to change the "permissions" of a directory or file. There are three kinds of permissions that may be assigned singly or in any combination to directories or files: "read" (you can display the contents of the directory or file), "write" (you can create, modify, or delete the directory or file), and "execute" (you can run programs). To fine tune the property structure even further, Unix allows owners to assign different combinations of the above permissions to the "owner" (the user), the user's assigned "group," and all "others".

The total grid of property relations is symbolized in the notation displayed in the leftmost column of the "ls -l" or long listing display of a directory's contents (on the ls command, see §5.2). Take the example of the following long listing of a directory:

humanitas> ls -l					
total 32					
drwx-----	2 ayliu	512	May 24 00:22	News/	
-rw-----	1 ayliu	13073	Jan 2 1993	agrippa	
-rw-----	1 ayliu	1919	May 25 21:07	d2	
drwxr-x---	2 ayliu	512	Sep 9 1993	for_all_eyes/	
drwxrwx---	2 ayliu	512	May 22 18:14	for_all_hands/	
-rw-----	1 ayliu	3372	Jan 28 16:38	info	
drwx-----	2 ayliu	512	May 24 21:27	mail/	
-rw-----	1 ayliu	8729	Feb 9 23:13	web	

The leftmost column consists of spaces for ten letters or blanks (" "). The first space designates whether the item listed is a directory ("d") or file ("-"). Then there are nine spaces organized into three triplets. The first triplet describes the permissions for the "owner"; the second for the "group"; and the third for "others." In these triplets, the first space is "r" for "read" (or blank), the second "w" for "write" (or blank), and the third "x" for execute (or blank). Thus in the above example, the "News/" directory is described as "drwx-----" (meaning this is a directory for which the owner has all permissions and everyone else has none). Or again, the file "agrippa" is described as "-rw-----" (the owner can read and write this file while everyone else is barred).

By default, your home directory has "drwx-----" permission. Only you can list its contents or do anything with any subdirectories or files it contains. Any subdirectory you create also has "drwx-----" permission; while any file you create in your directories has "-rw-----" permission (only you can read and write; if you create a program or shell script, you will need to add an "x" to make it executable).

For most users, the maximum privacy afforded by the above defaults is perfect. However, you may on occasion wish

to make a directory or file accessible to other Humanitas users. For example, I keep a public-access "for_all_hands" subdirectory in my home directory with files that other Humanitas users can read or copy (but not modify or delete). To make a file accessible to others, the owner has to change not only its permissions but the permissions of its containing directory and any higher-level directories so that others have a clear "path" all the way through. Files in my for_all_hands directory thus have "-rwxr-xr-x" permission (I have full rights; others can read and execute but not modify); the for-all-hands directory itself has "drwxrwxrwx" permissions (full access to anyone); and my top-level /home/ayliu directory has "drwxr-xr-x" permissions (others can list the filenames it contains but can't read the actual files unless I also change the permissions for individual files; no one else can modify my files).

7.2: CHANGING PERMISSIONS WITH THE `chmod` COMMAND

To change the permissions of a directory or file, use the command: "`chmod [permission-code] [directory or file name]`". The permission code can be specified either according to an alphabetical or numerical scheme. Only numerics are covered here. A numeric permission code consists of three digits, the first for the owner, the second for the group, and the third for others. The possible digits are 0-7:

0 = "---"

1 = "--x"

2 = "w--"

3 = "-wx"

4 = "r--"

5 = "r-x"

6 = "rw-"

7 = "rwx"

Examples speak clearest here. Let's say I wish to change the permissions of the file "agrippa" so that I have full access but everyone else can only read without modifying. I would use the command: "chmod 744 agrippa".

Again, to change the permissions of the file so that everyone has full access, I would use "chmod 777 agrippa".

🐼 Advanced users will want to edit their .cshrc system file to configure Unix to their liking. First copy .cshrc to a different name for safety's sake. Then call it up in Pico and edit it to change or include "set" and "alias" statements. The "set" statement creates a variable that can be invoked at the humanitas> prompt by typing "\$[variable]". For example, the statement "set pub=/home/ayliu/for_all_hands" sets the variable "pub" for "/home/ayliu/for_all_hands". To change to the for_all_hands directory, all one would need to do at the humanitas> prompt is type: "cd \$pub". "Alias" statements create keystroke "macros" such that, for example, "alias la ls -a" makes "la" equivalent to "ls -a". All Humanitas users have a .cshrc file that comes with standard set and alias statements. After a bit of tinkering, my own .cshrc presently looks like this:

Note: an edited `.cshrc` file will not take effect until the next time you logon to Humanitas.

☛ Advanced users may also want to use the `.history` system file. This file keeps a list of recent commands issued at the `humanitas>` prompt. Typing "**history**" displays this list with a reference number in front of each past command. Typing "**![number]**" reissues the specified command. Typing "**!!**" repeats the most recent past command.

```
alias hi history
alias jo jobs
alias ll ls -lsg
alias ls ls -F
alias lt ls -lt
alias lr ls -R
alias la ls -a -s
alias lp 'pwd; ls -a -s'
alias m more
alias nn nn -g -r
alias po popd
alias pu pushd
alias rm 'rm -i'
alias so source
alias z suspend
alias clear
endif
```

PART III

THE INTERNET

CHAPTER 8:

OVERVIEW OF THE INTERNET

8.1: CAPSULE HISTORY:

The Internet is the global "network of networks" or meta-network that currently links over 2,217,000 host or server computers around the world via hardwire, satellite, and other communication channels.* Most of these nodes serve multiple, local computers.

Initiated in 1969 with just four nodes (at UCLA, Stanford, University of Utah, and UCSB), the Internet--then known as ARPAnet--was originally developed under Pentagon sponsorship in part to sustain U. S. computer connectivity in the event of a nuclear strike. The design specifications called for it to connect point A to Z over just about any mismatched assortment of intervening hardware and software routes; and to do so flexibly such that the destruction of some routes would not block data getting through by other means. In the beginning, that is, the Internet was what contemporary cinema calls "post-apocalypse." It was the information highway for the road warrior.

How to ensure connectivity amid the hypothetical rubble? The elegant savagery of the solution parallels "a thousand plateaus," "lines of

* As of May 1994; see Susan Calcari, "A Snapshot of the Internet," Internet World 5, no. 6 (Sept. 1994): 54. It should be pointed out that "global" is not quite right; Internet presence in Africa, the Far East, and parts of South America are thin or non-existent. See the map in Daniel P. Dern, The Internet Guide for New Users (New York: McGraw-Hill, 1994), p. 17.

flight," "multiplicities," "schizophrenia," "packs," "many wolves," "bodies without organs," and all the rest of the screaming, postmodern philosophical horde.** The solution lay in "packet-switching." In packet-switched data transmission (later standardized in the TCP/IP protocol that is the circulatory system of the present Internet), messages are broken up into discrete "packets" each prefaced by a header identifying its origin, destination, and sequential position in the overall message. Then the packets are released like a flock of carrier pigeons to find their own way across a patchwork of nodes each of which assigns routing directions "on the fly." (As it were: "Weather advisory: heavy network traffic in the north; fly south." Or again: "nuclear strike in Chicago; go through Dallas.") To stay with the metaphor: some packets fly north, some south; some make a beeline, some get tired and rest for a while. But in the end (usually), all arrive at the destination and are reassembled in their original order.

As evidenced by the logarithmic increase in nodes, users, and uses during the '70's and '80's, the Internet concept was highly successful--so successful, indeed, that it rapidly outgrew its original military framework (a separate military MILnet split off in 1983). In 1989, the National Science Foundation (NSF) took over sponsorship of the Internet and initiated a series of bandwidth and speed upgrades to the national linkway or "backbone." Participation by educational, government, and other such institutions was subsidized on the understanding (formalized in an AUP or Accepted Use Policy) that the Internet could not be used for commercial purposes.

** It may be noted that the inception of the Internet in 1969 coincides roughly with the intellectual ferment touched off by "May 1968" in Paris; the phrases in quotes are from Gilles Deleuze and Félix Guattari, A Thousand Plateaus: Capitalism and Schizophrenia, trans. Brian Massumi (Minneapolis: Univ. of Minnesota Press, 1987).

By the 1990's, however, the business world wanted in as well. Much of the current story of the Internet, therefore, has to do not only with its outreach to such previously underconnected portions of the research-and-education community as humanities departments and undergraduates but with the creation of CIX, CommerceNet, MecklerWeb, and other commercial extensions of the Net. These "net profit" universes, as it were, are hard at work trying to adapt the decentralized, "many to many" pragmatics of the Net (see below) to the traditional "one to many" structures of mass-marketing (and vice versa). Such mainstreaming has been paralleled by a frenzy of Internet coverage in newspapers, TV, and other conventional one-to-many broadcast media. As witnessed in the salacious opportunism of the 1994 Los Angeles Times "scoop" on pornographic graphics files secreted in a publicly-funded research computer, or again--with a different kind of lust--in the recent, wall-to-wall press coverage of telecommunication and cable company mergers promising the proverbial "500 channels of on-demand entertainment," the one-to-many media does not yet know quite what to make of the legitimacy of many-to-many networking. Either the Net is all pornography or it is the corporate promised land.

As of 1994, the NSF is officially ceding Internet sponsorship to the private sector and retiring its no-commercial-use rule. Concomitantly, therefore, the Internet has suddenly begun to worry about such prerequisites for doing business online as "authentication," "encryption," and "pricing structure."

8.2: GENERAL PRINCIPLES

While it is possible that post-Cold-War commercialization will eventually make obsolete the character the Internet has taken on from the originally strangelove imagination of the military-government-academic

communities, be prepared in the meantime for the following general principles of the Net:

1. *There is no one in control* (but see principle 5 below). The Internet is a confederation of millions of computers and users spread across institutional, state, and national boundaries. If someone puts a program or resource on the Net that others like, it propagates outwards like a "good" virus (as happened with the invention of Gopher in 1991 or of the World Wide Web in 1992). No one decrees it; no one censors it. The best analogue for the Internet is a living organism at the evolutionary level, say, of a fungus. Insofar as there is a government of the Net, it is government by convention (codified in Accepted Use Policies and "Netiquettes"). If you do something really unmannered or selfish (like the infamous law firm that recently posted an advertisement on thousands of Usenet newsgroups), whole mobs will variously "flame" you, forge cancellations of your postings, choke your local system with e-mail, notify your system administrator, or otherwise throw electronic eggs on your door. Until recently, even simple ignorance could earn you a scarlet letter. But the recent Net-arrival of hundreds of thousands of "newbies" (including undergraduates, humanities types, and the infamous AOL'ers or America Online subscribers) is beginning to tip the playing field.

2. *Many speak to many.* In the mainstream media, one (or, at most, two or three) broadcasts to the masses. Thus there are traditionally three network TV channels, one cable provider, one local newspaper, two weekly news magazines, one MTV (VH-1 doesn't count), etc. By contrast, the Internet is many-to-many. Because overhead is low compared to traditional media, there are many thousands of "broadcasters" on the Net (FTP sites, Gopher sites, Web sites, etc.) as well as millions of what might be called micro-broadcasters (individuals posting by e-mail or Usenet). Content, as opposed to start-up capital, is the critical factor in finding a broadcast audience (a levelling phenomenon that recently had a commercial giant like MTV suing an individual--ex-MTV VJ Adam Curry--over his claim to an Internet site-name beginning with the letters, "mtv"). Or

more accurately, because the accent is on full interactivity, the very notion of "broadcast" is out of joint. The Net is more like a gigantic bulletin board or switchboard. Whereas interactivity for cable TV might mean the home-shopper presses one of three buttons on the remote, for the Internet it means the same shopper gets to post a detailed, expletive-undeleted critique of the product read by multitudes.

3. *Things change.* A common experience on the Net is to ftp, gopher, or mosaic to a site one has visited yesterday only to find that it has (a) disappeared, (b) been taken temporarily off-line, (c) become so overloaded it won't admit outside users during daylight hours, or (d) changed its organization and contents. Everything is in motion; nothing is at rest. Thus while it is tempting for humanities users in the academy to conceive of the Internet on the model of a giant library, the model does not really fit. Library users have some assurance that once a book has been published and acquired it can be found in the same form at any future date (either locally or through Interlibrary). On the Net, however, nothing guarantees that files will remain in existence, stay in the same form, or retain the same name. Many sites and Usenet groups hold material in "archives"; but then the archives themselves come and go.

4. *No one knows what it costs.* There have been estimates that the maintenance and development of the Net (not including the expense of local systems) currently costs about \$.50 per user per month. However, there is no rational link between such estimates and what the individual user perceives on the bottom line because there is at present no rational "metering" system for access, bandwidth, or content. (a) A large proportion of the Net population get access through subsidized university, government, or other such institutions, with the result that access appears "free." (b) Access through such commercial providers as America Online, Delphi, Netcom, CompuServe etc., is definitely not free; but rates and pricing structures vary because there is no common paradigm for what exactly users are paying for (time on the commercial service, time on the Internet as well as time on the service, downloading time, number of

e-mail messages per month, etc.?). (c) Irrespective of whether a user has "free" or commercial access, bandwidth-usage is heedless of elementary laws of supply and demand. Except in the matter of phone charges (plus any flat-rate or per-hour charge levied by a commercial provider), it apparently makes no difference whether a user posts one message on Usenet or a thousand. Or again, asking for a one-page document to be sent via ftp from a remote site costs the same as asking for all of Shakespeare's works. (d) At the foundation of the above pricing chaos, it has been speculated, are two forms of "dark matter" on the Net resistant to accounting (alluding to the so called dark or "missing" matter that physicists calculate must exist somewhere in the universe). One is the "dark bandwidth" of currently underutilized telecommunications optical cable. The other is the "for free" ethos of freeware and shareware that dominates the Net. Many, many programmers, system administrators, and others contribute "extra" time and expertise for no visible remuneration. The Internet may not be Gospel, but it has its proselytizers.

What this all means for the individual user is that not even the invisible hand is in control of the Net. There is no way to pay a premium, for example, to get priority access to a popular ftp or telnet site. Instead, the only metering system in place at such sites is a brutal On/Off switch. When too many freeloaders try to get on board, the site turns away everyone after the first ten or twenty.

5. *Because no one, not even the invisible hand, is in control, the Sysop is in control.* This seems to contradict principle 1 above, but it is actually a corollary. Because no one governs the Internet as a whole, the local Sysop or system administrator governs your access and your system totally.

Think of the Sysop as a combination mayor and local sheriff in the old West. He (in our case, she) runs the town as efficiently, fairly, and progressively as possible. But she's always on the lookout for bandits from out of town (hackers); and she's just as hard on offenders who live in town. There is no formal appeal, checks-and-balances, or oversight committee. Some local systems publish Accepted Use Policies or other rules that users

are expected to know. But presently neither Humanitas nor UCSB as a whole has a declared Net policy. (It will be interesting, for example, if there is every a legal question regarding what is acceptable Net usage for a UCSB user and whether the university or state has the right to look at that individual's e-mail.)

8.3: THE INTERFACE

Where Byron's Don Juan began, "I want a hero," anyone beginning on the Internet might say, "I want a metaphor." There is no conceiving the Net without a metaphor. Thus, for example, the multitude of metaphors built into the Net itself to describe its structure--e.g., "web," "mosaic," "gopher tunnel," and, of course, "net."

This manual will adopt a metaphor centering on how the Net appears to us through its interfaces. Imagine that our combined interface to the Net is a windshield. More specifically: the windshield of something like a recent jet fighter with "heads-up display" (featuring indicators, targeting windows, and other telltales projected over the landscape in front of the pilot's eyes) (figure 1).

The landscape that appears through the windshield on the horizon is the global Internet, whose main features include ftp sites (read-only archives of files that can be browsed or downloaded), telnet sites (host computers that allow remote users to log on and use certain commands; e.g., most library on-line catalogues), Usenet newsgroups (approximately 7,000 special-interest bulletin boards where users can read and post messages), Gopher sites (servers that organize files for browsing or downloading according to hierarchical menus), World Wide Web sites (servers that store a special kinds of files that include hypertextual, multimedia links to other files), and, of course, servers through which users can address e-mail to other individual users.

The landscape closer to hand is the UCSB campus "backbone" that links all our local servers together and feeds in turn to the Internet. Humanitas, for example, is just one of our campus's servers, which are specialized by field (e.g., Humanitas for the humanities, Alishaw for the social sciences, etc.) or task (e.g., Mercury currently handles the Usenet "newsfeed" for Humanitas).

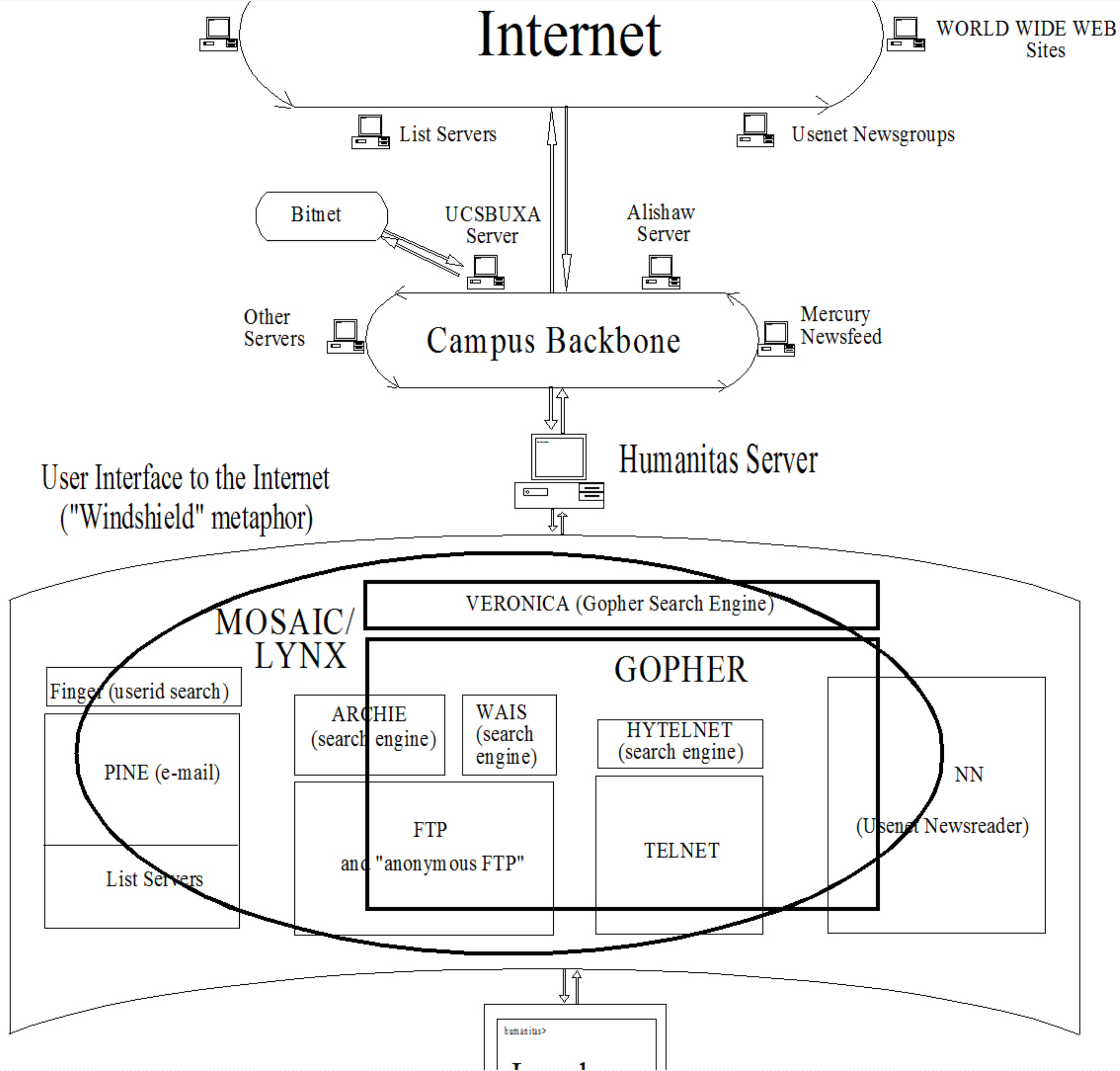
Most of this total, electronic terrain functions according to a "server/client" paradigm of distributed computing. This means, in essence, that remote sites do not simply pour content into local servers and/or user computers as if the latter were passive containers. Rather, local "client" programs cooperate actively with remote servers to initiate, coordinate, and organize data requests and transmissions. (Thus, for example, there is a Gopher client-program running on Humanitas to collaborate with remote Gopher servers.) The interaction of each kind of local client with its matching remote server generates an interface or phenomenal manifestation particular to that part of the Internet landscape. Gopher, ftp, the Web, and so on, for example, each organize the user's computer screen in different ways and offer different commands. Combining all these individual interfaces together in our mind's eye (unless one has an X-windows system running on a powerful workstation, the interfaces never actually appear together) produces the overall Internet interface figured in the heads-up "windshield" in **Figure 1**.

In its totality, this windshield will at first seem pure hodge-podge. As might be predicted from the Net principles outlined earlier, the interface to the Net is not one view or even one logical suite of views. Rather, it is a kaleidoscope or stained-glass window: some panes show just a tiny sliver of the terrain in one direction, others like Gopher and Mosaic are more inclusive, yet add idiosyncracies of their own.

There are two, general patterns to the fracture, however. First, there is a rough progression we can follow from older, more elementary interfaces with strictly-defined uses to newer, higher-level interfaces with

flexible or generalized uses. And secondly, the most important interfaces tend to be matched up with their own "search engines."

In the following descriptions, I move approximately from the old to the new and the elementary to the generalized, stopping to consider associated search engines as appropriate.



CHAPTER 9:

E-MAIL (AND FINGER SEARCH ENGINE)

9.1: PINE E-MAIL

Since the first days of the Internet, when it was found that the bulk of traffic on ARPANet consisted of personal messages, e-mail has been the most popular Net application. There are several ways to read, send, and manage your e-mail. Most directly, the **mail** command in Unix (together with its options and various "redirection" commands) gets the job done. However, using **mail** is like marrying the mailman or mailwoman: you gain a lot of control, but you have to snuggle up to zip codes, routes, etc. Most users will therefore want to use such "friendly," menu-driven e-mail interfaces available on Humanitas as Elm and Pine.

For the average user, Pine (currently version 3.89 on Humanitas) is by far the best choice.

9.2: PINE MAIN MENU SCREEN

(STARTING PINE)

To start pine, type "**pine**" at the humanitas> prompt. By default, Pine's Main Menu will come up with a highlight bar positioned over the Folder List option as follows:

PINE 3.89	MAIN MENU	Folder: INBOX	12 Messages
?	HELP	-	Get help using Pine

C	COMPOSE MESSAGE	-	Compose and send a message
I	FOLDER INDEX	-	View messages in current folder
L	FOLDER LIST	-	Select a folder to view
A	ADDRESS BOOK	-	Update address book
S	SETUP	-	Configure or update Pine
Q	QUIT	-	Exit the Pine program

Copyright 1989-1993. PINE is a trademark of the University of Washington.

? Help	P PrevCmd	R RelNotes
O OTHER CMDS	L [ListFldrs]	N NextCmd
		KBlock

All Pine screens include a status line at the top. The status line for the Main Menu shows the current active folder ("Inbox" when Pine starts) and the number of messages in that folder. Pine screens also include at the bottom a two-line menu of commands relevant to the particular screen. Where there are more commands than can fit on two lines, the "O OTHER CMDS" option cycles through the whole repertory. If you configure your personal .pinerc file so that you are no longer a "seedling"-user but a "sapling" or "old-growth," the bottom-screen menu will show a progressively greater number of menu options. (Created at the Univ. of Washington deep in the heart of evergreen country, Pine is intent on letting you know that using e-mail saves trees.) The examples shown in this manual, for example, are "old-growth." Some of the

fancier options are not yet "enabled" in the current version of Pine but will come alive in future releases.

To invoke any Pine command, simply type its menu letter. (You can use lower-case despite the fact that letters are shown in upper-case on the screen.) For example, typing "l" while in the Main Menu moves you to the Folder List screen. You can use any command relevant to your current Pine screen whether or not its menu letter presently shows at the bottom of the screen. For example, typing "c" from anywhere in the Main Menu, Folder Index, or Message Text screens puts you in the Compose Message screen even if the "c" menu item is not visible.

Besides its status line and bottom-screen menu, the Main Menu screen promenades its most crucial commands on center screen complete with a highlight bar you can move with your arrow keys. Pressing <Return> when the bar is over an option triggers that option. To get back to the Main Menu from any subsequent Pine screen, press "m" for Main Menu. To quit Pine, press "q" from any screen.

```
feature-list=old-growth,  
expunge-without-confirm
```

The initial keystroke list allows you to create a "macro" instructing Pine to go through an automatic sequence of keys each time it starts up. For example, completing the line as follows causes Pine to begin not in the Main Menu but in the Folder Index screen with the highlight bar positioned over the first new message in your Inbox:

```
initial-keystroke-list=i
```

To browse your .pinerc file, go "more .pinerc" or "pg .pinerc."

9.3: FOLDER INDEX SCREEN

(SEEING THE LIST OF MESSAGES IN YOUR INBOX)

Pine users will usually first want to move from the Main Menu to the index of files in their Inbox. While in the Main Menu, type "i" to bring up the Folder Index screen for the default folder (Inbox). The following is an example (but with the "Re:" text blanked out to protect the privacy of my correspondents):

+	1	Nov 11	Tom Dowe	(12,239)	Re: xxxxxxxxxxxxxxxxx
+	2	Aug 8	CLYMER@BINAH.CC.BR	(1,478)	Re: xxxxxxxx
+	3	Aug 8	Terri Jo Ortega	(1,615)	Re: xxxxxxxxxxxxxxx
+ A	4	Aug 9	Terri Jo Ortega	(4,447)	Re: xxxxxxxxxxxxxxxxx
+ A	5	Aug 9	Albert S. Lindemann	(1,110)	Re: xxxxxxxxxxxxxxxxx
+ A	6	Aug 8	Terri Jo Ortega	(4,309)	Re: xxxxxxxxxxxxxxx
+	7	Aug 26	Rita Raley	(4,765)	Re: xxxxxxxxxxxxxxxxx
+	8	Aug 26	Vince Willoughby	(1,615)	Re: xxxxxxxxxxxxxxx
D	9	Aug 28	Stuart Curran	(1,876)	Re: xxxxxx
D	10	Sept 2	frank murray	(4,960)	Re: xxxxxxxxxxxxxxxxx
+ N	11	Sept 3	Terri Jo Ortega	(1,780)	Re: xxxxxxxxxxxxxxxxx
N	12	Sept 3	frank murray	(6,675)	Re: xxxxxxxxxxxxxxxxx

? Help	Q Quit	L ListFldrs	Z Zoom	Y Print	S Save
O OTHER CMDS	C Compose	G GotoFldr	W WhereIs	T TakeAddr	E Export

Each line in on the index screen gives identifying information about one e-mail message you have received: the date the message was sent, the sender, the size of the message (in number of characters), and the topic. A "+" at the left means that the message was sent to you as an individual rather than to a cc: or listserver group. "N" means that the message is new and you have never looked at it. "A" means that you have already answered this message by using the "reply" feature.

"D" means that the message has been marked for deletion and will not reappear in your Inbox the next time you open up Pine.

(Any message you have received will remain in your Inbox folder, and will thus continue to show up on the index screen, until you take positive action to delete it or save it to a folder other than Inbox.)

When the index screen opens, the highlight bar will be positioned over the first new message in Inbox. You can move the highlight bar up and down with your arrow keys, or (if you are "old growth") type a number followed by <Return> to jump the bar to the appropriate message. Pressing <Return> once more will then display the selected message.

Useful commands in the index screen include the following:

- ▶ *"d"* marks the message at the highlight bar for deletion (by default, Pine will ask upon quitting to confirm that you wish to delete marked messages).
- ▶ *"u"* undeletes a file marked for deletion (this must be done before quitting Pine or the file will be gone)
- ▶ *"w"* allows you to search for a key word on the index screen. For example, typing *"w"* and then, when prompted, *"dowe"* would jump the highlight bar to the file whose index line contains the word *"dowe."*
- ▶ *"f"* forwards a message to another correspondent.
Pressing *"f"* puts you in the Compose Message screen with the Subject line preset to that of the original message. The text of the original message appears on the screen under the line, *"-----Forwarded message-----"*. Just fill out the *"To:"* line in the header to your message, add any comments you wish (such as, *"John, this is something I picked up from a listserv group that you might be interested in"*), and send the message off.
- ▶ *"s"* saves the message to another folder managed by Pine. When you press *"s"*, a prompt will appear

indicating that the selected message will be saved by default to a folder named "saved-messages". But you can substitute the name of any other folder you use to archive your e-mail (e.g., folders named by topic, by correspondent). If the folder does not already exist, Pine prompts to confirm that you wish to create it. Once it has been saved in this fashion, the original message in the Inbox folder is automatically marked for deletion.

- ▶ "e" exports (i.e., writes) the message to a Unix file in your storage space so it can be downloaded or otherwise manipulated independently of Pine. After you press "e", Pine will prompt for a filename and then write the file to your home directory. By prefixing a path name, you can also write to a different directory. If the filename you specify already exists, Pine asks whether you want to replace or append to it. The original e-mail message, once exported, is retained without being automatically marked for deletion. (Note: most Pine "folders" are themselves actually Unix files located in the "mail/" subdirectory of your home directory. If you look at any of these files in Unix using **more** or **pg**, you will see that they contain your saved messages collected one after the other. But the Inbox folder is exceptional in that it cannot be accessed directly from Unix unless you export its contents.)

9.4: MESSAGE TEXT SCREEN

(READING A MESSAGE)

To read a message, position the highlight bar over any item in your folder index and press <Return>. The selected message will be displayed in Pine's Message Text screen as follows:

PINE 3.89 MESSAGE TEXT	Folder: INBOX Message 11 of 12 100%																		
<p> Date: Sat, 3 Sep 1994 10:17:24 -0700 (PDT) From: Terri Jo Ortega <tjort@humanitas.ucsb.edu> To: Alan Liu <ayliu@humanitas.ucsb.edu> Subject: gopher help files </p> <p> Alan, Try the gopher help now, I think I fixed the problem. </p> <p> TJ </p> <p> On Friday, 2 Sept 1994, Alan Liu wrote: </p> <p> >TJ, >The help file for Gopher is dead. Invoking the "help" option from the Gopher >screen brings up an error message saying that the help file could not be found by >the system in the expected directory. > </p>																			
<table style="width: 100%; border: none;"> <tr> <td style="border: none;">?</td> <td style="border: none;">-</td> <td colspan="4" style="border: none;"></td> </tr> <tr> <td style="border: none;">? Help</td> <td style="border: none;">M Main Menu</td> <td style="border: none;">P PrevMsg</td> <td style="border: none;">- PrevPage</td> <td style="border: none;">D Delete</td> <td style="border: none;">R Reply</td> </tr> <tr> <td style="border: none;">O OTHER CMDS</td> <td style="border: none;">V ViewAttch</td> <td style="border: none;">N NextMsg</td> <td style="border: none;">Spc NextPage</td> <td style="border: none;">U Undelete</td> <td style="border: none;">F Forward</td> </tr> </table>		?	-					? Help	M Main Menu	P PrevMsg	- PrevPage	D Delete	R Reply	O OTHER CMDS	V ViewAttch	N NextMsg	Spc NextPage	U Undelete	F Forward
?	-																		
? Help	M Main Menu	P PrevMsg	- PrevPage	D Delete	R Reply														
O OTHER CMDS	V ViewAttch	N NextMsg	Spc NextPage	U Undelete	F Forward														

The status line at the top indicates which message is being displayed in the current folder (out of how many) as well as roughly where you are in the message ("100%" means you are seeing all the message; "67%" would mean that the last line on your present screen is 67% of the way into the message).

The header to the message displays the date and time the message was sent, the name and e-mail address of the sender, the primary recipient, who it was "cc:'d" to, and the subject of the message. "Parts/Attachments" appears at the end of the header information if the e-mail message has a file "attached" to it (see below).

The message text then occupies the main part of the screen. Any lines marked at the left by a ">" are text your correspondent has quoted by responding to some previous message with the "reply" command.

To scroll screen by screen through a message, use the spacebar. To scroll backwards, use the "-" (hyphen) key. You can also scroll a line at a time with the up and down arrow keys.

Useful commands in the Message Text screen include the following:

- "n" displays the next message in your inbox as listed in the sequence on the Folders Index screen.
- "p" displays the previous message.
- "i" returns you to the index of files in the Folder Index screen.

- ▶ *"d" marks the message currently being displayed for deletion when you close Pine.*
- ▶ *"u" undeletes a file marked for deletion (this must be done before quitting Pine or the file will be gone).*
- ▶ *"r" allows you to reply to the message currently being displayed. When you press "r", Pine will ask whether you wish to "Include original message in Reply (y/n/^C) [n]". "y" copies your correspondent's message and marks it with a row of ">"s down the left margin. (Use the pico editing commands to reduce the resulting quote to just the portion you want.) The default is "n" for no. If the original message has more than one recipient or a cc: list, Pine will also ask if you wish to reply to everyone. Be sure to answer "no" unless you really do wish to broadcast your reply. Once past this series of prompts, Pine puts you in the Compose Message screen with the header preset to your correspondent's e-mail address and the original subject of discussion. (See below for further details). Pressing "^C" (<Control> + C) at any time cancels the reply process.*
- ▶ *"f" forwards the message being displayed to another correspondent. Pressing "f" puts you in the Compose Message screen with the Subject line set to that of the original message. The text of that message (complete with header information) then appears on the screen under the line, "-----Forwarded message-----". Just fill out the "To:" line in the header to your message,*

add any comments you wish before the text, and send the message off.

- ▶ *"s"* saves the message being displayed to another folder managed by Pine. When you press *"s"*, a prompt will appear indicating that the selected message will be saved by default to a folder named *"saved-messages"*. But you can substitute the name of any other folder you use to archive your e-mail (e.g., folders named by topic, by correspondent, etc.). If the folder does not already exist, Pine prompts to confirm that you wish to create it. When the save operation is complete, the message is automatically marked for deletion from its original folder.
- ▶ *"e"* exports (i.e., writes) the message to a Unix file in your storage space that can be downloaded or otherwise manipulated independently of Pine. After you press *"e"*, Pine will prompt for a filename and then write the file to your home directory. By prefixing a path name, you can also write to a different directory. If the filename you specify already exists, Pine asks whether you want to replace or append to it. Once exported, the original message is retained without being automatically marked for deletion.
- ▶ *"t"* takes the sender's e-mail address in the currently displayed message and adds it to your Addressbook. Pine will prompt you for a nickname, full name, and e-mail address (the full name and address will

already be filled in by Pine, though you can alter them before saving the information).

- ▶ *"w"* searches for a keyword in the currently displayed message.
- ▶ *"v"* views any undisplayed "attachments" sent with the message. (If there are such attachments, they will be listed by number and brief description in a separate section of the message header called "Parts/Attachments.") When you press *"v"*, Pine will ask you to "Enter attachment number to view or save (1 - [highest number of attachments].)" Attachment number 1 is simply what you are already seeing on the screen. So if there is just one undisplayed attachment, you will want to press *"2"*. Pine will then ask, "Save or View attachment 2 ? (s/v) [s]". The default is to "save," which in this context means to export or write the file to Unix in your storage space (Pine supplies a filename from the original attachment, but will allow you to change it). The "view" option only works if the attachment is text-only.

Parts/attachments

1	Shown	9 lines	Text
2	OK	300 lines	Text, "William Gibson's _Agrippa_"

Number 1 is the e-mail message itself. Number 2 is the attachment, whose length, file type, and short description are listed. Use the "view attachments" command in the Message Text screen to save the attachment to a Unix file or to view it (if it is a text-only file). 🐼 Note that you can experiment with such relatively complex aspects of e-mail as "attachments" without fear of making guinea pigs out of your correspondents by sending test messages to yourself. Just address the message to your own e-mail address, send it, and it will show up as new mail in a few minutes.

9.5: COMPOSE MESSAGE SCREEN

(WRITING AND SENDING A MESSAGE)

To compose a message and send it to someone, type "c" from any Pine screen--whether Main Menu, Folder Index, List Folders, or Addressbook. You will be placed in the Compose Message screen with your cursor on the "To:" line in the header as follows. (If you start from Addressbook, Pine fills in the "To:" line in the header with the addressee under the highlight bar in Addressbook; similarly, if you reach the Compose Message

screen by way of the "reply" function described above, Pine fills in the "To:" and "Subject:" lines from the message you are replying to.) Note that the bottom-screen menu of commands in the Compose Message screen changes slightly depending on whether your cursor is in the header or in the actual message area:

PINE 3.89

COMPOSE TEXT

Folder: INBOX

12 Messages

To

Cc

Attchmnt:

Subject

^G Get Help

^C Cancel

^R Rich Hdr

^K Del Line

^O Postpone

^X Send

^D Del Char

^J Attach

^U UnDel Line

^T To AddrBk

On the "To:" line, fill in the e-mail address of your correspondent. This can be done in several ways. A local user on the Humanitas system can be addressed simply by userid (e.g., "ayliu"). Users elsewhere on the internet must be addressed using a full host and domain address (e.g., "ayliu@humanitas.ucsb.edu"). Or you can enter a nickname for a person previously registered in your Addressbook (e.g., "alan" if you have previously nicknamed ayliu@humanitas.ucsb.edu as "alan"). Also, "^T" puts you temporarily in the Addressbook screen, from which you can use the "s" command to select a name and insert it on the "To:" line in the message you are composing. (You may have to press the spacebar or some other key after "^T" if your system sticks at this point.) To address multiple correspondents, insert their addresses or nicknames on the same line separated by a comma and space: e.g., "alan, bill, julie". Pine will expand local or nicknames to full names once you press <Return>. For example, "bill" might become Bill Gates <bgates@microsoft.com>".

On the "Cc:" line, add any names you wish to send a "carbon copy" to. ("^R" expands the header section to add a "Bcc:" line allowing for blind carbon copies; once you have filled out the "Bcc:" line, pressing "^R" again hides the line from view so that the cc'd parties will not know who else was cc'd.) You can use all the options described under "To:" above to address the Cc: line. Again, to include multiple recipients list the names separated by a comma and space.

The "Attchmnt:" line can be filled in by pressing "^J" from anywhere in the header section (the cursor need not be on the "Attchmnt:" line itself). "^J" prompts you for the name of a Unix file to attach (see box above on attachments) as well as for a descriptive comment. When you are done, you will see the "Attchmnt:" line filled out with something like this:

`"/home/ayliu/agrippa (13KB) 'Text of Gibson's poem'".` Note: the advantage of putting the cursor on the `"Attchmnt:"` line when working with attachments is that the `"^T"` option on the bottom-screen menu is changed to `"To Files."` This allows you to bring up a list of the Unix files in your home directory together with a simple menu of file-management commands (`"s"` on this menu selects a file for attachment).

On the `"Subject:"` line, add a descriptive comment.

When done with the header information, hit `<Return>` or use the down arrow key to move your cursor into the message screen proper. (If you change your mind about the header information, you can always use the up arrow key to back up to the appropriate lines and edit, add to, or delete them.)

Once you are in the message screen, compose your message. Because by default Pine uses the Pico text editor for composition, you will see the following, familiar bottom-screen menu at this point:

<code>^G Get Help</code>	<code>^C Cancel</code>	<code>^R Read File</code>	<code>^Y Prev Pg</code>	<code>^K Cut Text</code>	<code>^O Postpone</code>
<code>^X Send</code>	<code>^J Justify</code>	<code>^W Where is</code>	<code>^V Next Pg</code>	<code>^U UnCut Text</code>	<code>^T To Spell</code>

See § 6.2 for details on editing commands in Pico. Note especially the following editing options:

- You can move or copy text using the `"cut"` (delete) and `"uncut"` (undelete) commands. To move text, go `"^K"` to cut or kill a line of text; move the cursor to the desired position; then go `"^U"` to uncut the last cut line or last consecutive series of cut lines. The deleted material will be restored at the cursor position. To copy text, follow the above procedure but first `"uncut"` the deleted material at its original

position before uncutting it again at a new cursor position (deleted text is placed in a buffer and can be repeatedly uncut until replaced by new deleted text).

To make "**^K**" and "**^U**" operate on a block of text rather than a line or lines, use "**^^**" (i.e., **<Control> + ^**) to begin a highlighted block. Move the cursor to the end of the desired block, then use "**^K**" and "**^U**" as appropriate.

- ▶ "**J**" reformats the paragraph you are working on from the line at the cursor onward. This is useful when you have created irregular lines by deleting or adding material. It is also useful when you have imported a file with long text lines that would otherwise be cut off at the right of the screen (e.g., an ASCII file descended originally from a word processor document with small or proportional fonts).
- ▶ "**^R**" reads or imports a Unix file into the e-mail message you are composing. This is one of the most powerful commands in Pine, since it allows you to create a long or complex document on your word processor, upload it in ASCII format to Humanitas as a Unix file (add an extra, blank line at the end), and then import it into Pine for transmission. When you press "**^R**", Pine prompts you for the name of the file to import. (Note that your cursor has to be in the message text area of the screen to use "**^R**" in this way. While in the header area, "**^R**" has a different meaning.)

- *"^C" cancels the message you are working on and returns you to whatever Pine screen you were in prior to entering the Compose Message screen. Before the cancellation occurs, you will be asked to confirm that you really mean to throw away what you have been writing.*

When you are done composing a message, "^X" sends it and places a copy in the Pine "sent-mail" folder. If you wish to suspend a message you are composing without sending it or erasing it, use the "^O" command. You can then move elsewhere in Pine or even quit Pine entirely; the next time you press "c" to enter the Compose Message screen, Pine will ask if you want to resume work on the suspended message. (Note: if you lose your phone link or your power goes out in the middle of composing a long letter, Pine usually "suspends" any message you were composing. It should be there waiting for you when you log on again.)

```

-----
-----
----- Dept. of
English -----
----- Univ. of
California -----
----- Santa Barbara, CA
93117 -----
----- FAX:
805/893-4622 -----
-----
-----

```

These signatures are added automatically to the bottom of e-mail messages and Usenet postings by means of a system file named ".signature" in your Unix home directory. If you want to create a signature, use pico to create ".signature". (You can glean interesting ASCII art ideas from the Usenet newsgroup,

9.6: FOLDER LIST SCREEN

(MANAGING FOLDERS)

Pressing "l" for "list" from anywhere in the Main Menu, Folder Index, or Message Text screens puts you in the Folder List screen, which simply displays the names of your folders.

Everyone has the folders: Inbox, sent-mail, and saved-messages.

You can also create other folders to archive messages. For example, my Folder List Screen currently looks like this:

PINE 3.89	FOLDER LIST	Folder: INBOX	12 Messages
INBOX	sent-mail	saved-messages	cyberpunk
e-romantics	pmc	sent-mail-aug-1994	

? Help	M Main Menu	P PrevFldr	- PrevPage	D Delete	R Rename
O OTHER CMDS	V [ViewFldr]	N NextFldr	Spc NextPage	A Add	

Use your arrow keys to move the highlight bar from folder to folder, then go <Return> to open up the index list for that folder. Once you have moved to a different folder in this way, your working location will be in that folder until you either change location again or quit and restart Pine (which automatically puts you in Inbox). The "A", "D", and "R" commands allow you to manage your folders.

Note that Pine periodically allows you to clear out your sent-mail folder, which would otherwise endlessly accumulate copies of messages you have sent. At the end of each month, Pine asks whether you wish to move the messages in sent-mail to, for example, sent-mail-aug-1994. You can answer "n" to expunge old sent mail. (Generally, it is a good idea to archive one previous month's worth of sent mail; delete older sent-mail folders to minimize the amount of space Pine occupies in your limited disk storage.)

9.7: ADDRESSBOOK SCREEN

(KEEPING A CONTACT LIST)

One of the most convenient features of Pine is that it allows you to keep an organized address book of correspondents. There are three ways to access your address book.

From the Main Menu screen, the "A" option puts you in Addressbook with a full range of management options as follows:

PINE 3.89 ADDRESS BOOK		Folder: INBOX	Message 11 of 12 NEW
porter	Abbott, Porter	pabbott@humanitas	
ann_adams	Adams, Ann Jensen	ajadams@humanitas.ucsb.e	
alix	Alix, David	9531alix@ucsbuxa	
allaback	Allaback, Steve	steveall@hcfmail	
ask	Ask Humanities Computing	askhcf@humanitas.ucsb.ed	
austin	Austin, Mike	austin@humanitas	
baldwin	Baldwin, Laura	lbaldwin@hcfmail	
laura	Baldwin, Laura	lbaldwin@humanitas	
bate	Bate, Jonathan	ajb6494@liverpool.ac.uk	
bender	Bender, John	jbender@leland.stanford.	
bialostosky	Bialostosky, Don	userrc49@umichum	
michelle	Bigos, Michelle	mbigos@hcfmail	
blakley	Blakley, Johanna	6500jb@ucsbuxa.ucsb.edu	
sheridan	Blau, Sherida	sblau@hcfmail	
bliss	Bliss, Lee	lbliss@hcfmail	
lee	Bliss, Lee	lbliss@hcfmail	
maurizia	Boscagli, Maurizia	boscagli@hcfmail	
brooks	Brooks, Dwight	brooks@humanitas	
brown	Brown, Marshall	mbrown@u.washington.edu	
? Help M MainMenu P PrevField - PrevPage D Delete S CreateList O OTHER CMDS E [Edit] N NextField Spc NextPage A Add Z AddToList			

Useful Addressbook management options include the following:

- "A" allows you to add new entries. Pine will prompt for a nickname (e.g., "porter"), a full name ("Abbott, Porter"), and an e-mail address ("pabbott@humanitas"). Note that you can set up multiple entries for the same person under different nicknames (e.g., "bliss," "lee" above). This is helpful when you are addressing a message in the Compose Message screen and can't remember what nickname

you have assigned a correspondent. Note also that local Humanitas users can be defined with an abbreviated e-mail address--e.g., "lbliss@hcfmail" ("hcfmail" is the older name for Humanitas; the system doesn't care whether you use "hcfmail" or "humanitas" in addresses).

- ▶ *"E"* edits the field under the highlight bar. In the example above, *"E"* edits the nickname, "porter."
- ▶ *"W"* (shown under "Other Commands") searches for a name or other keyword in the Addressbook.
- ▶ *"C"* puts you in the Compose Message screen with the "To:" line filled out with the entry under the highlight bar.
- ▶ *"S"* and *"Z"*. These options allow you to set up "lists" or groups of correspondents for collective mailings. To create a list, first add to Addressbook all individual entries for your correspondents, then go *"S"*. Pine will prompt you for a list title (e.g., "English Department Staff"), a list nickname (e.g., "staff"), and then the addresses of each of the list members (you can enter just their nicknames for convenience). *"Z"* subsequently allows you to add new members to the list. To use your list addresses, just fill out the "To:" or "Cc:" line in a message with the list nickname. "Staff", for example, automatically expands into all the members of my department staff.

From the Compose Message screen: Pressing "^T" while your cursor is on the "To:" or "Cc:" line of a message you are composing takes you temporarily to Addressbook with the following, more limited bottom-screen menu:

? Help	E ExitSelect	P PrevField	- PrevPage	
	S [Select]	N NextField	Spc NextPage	W WhereIs

Pressing "s" with the highlight bar positioned over the selected entry in your Addressbook returns you to the Compose Message screen with the header line properly addressed.

From the Message Text screen: While reading a message you have received, you can press "t" to "take" the address in the header and record it in Addressbook. In this case, you will not see the Addressbook screen itself. Instead, Pine prompts you for a nickname, full name, and address before reporting that it has updated Addressbook and then allowing you to continue reading your message (the full name and address are supplied by default from the message currently on the screen, but you can change them if you like).

9.8: FINGER SEARCH ENGINE

Associated with e-mail is an elementary search engine named Finger that helps you locate e-mail addresses and other pertinent information about Internet users. (There are a variety of more advanced "knowbots," "whois," and other address-list instruments available on the Internet, but they primarily serve the scientific, government, defense, and

commercial communities. Humanities scholars tend to be invisible to such instruments.)

To find information about a Humanitas user, type the following at the humanitas> prompt: "finger [first-name, last-name, or userid]". (A first or last name can be entered in either lower or upper case letters; userids, however, are case sensitive.) For example, a finger search for "liu" on the Humanitas system turns up the following:

```
humanitas> finger liu
Login name: ayliu                      In real life: Alan Liu
Directory: /home/ayliu                Shell: /bin/csh
On since Sep  7 10:17:05 on tty1 from 128.111.224.251
14 seconds Idle Time
Mail last read Wed Sep  7 12:12:19 1994
Project: Understanding what "literary" means in the age of e-mail.
Plan:
Address: Dept. of English, Univ. of Calif., Santa Barbara, CA 93106
Office:  South Hall 2526                Phone: (805) 893-4673
                                           Fax:   (805) 893-4622
                                           E-mail: ayliu@humanitas.ucsb.edu
```

The Login name is the userid that you would use to address an e-mail letter (e.g., "ayliu@humanitas.ucsb.edu"). The Directory is the user's home directory. Other pertinent (some think, impertinent) information then follows—including when a user last logged on, when he or she last opened up e-mail (you cannot actually tell, of course, whether a user "read" the mail or which mail was read); whether there is new mail waiting for the user to look at, etc. (Finger will also report if the user has never logged on.) The information displayed under the headings "Project" and "Plan" appear only if the user has created system files called ".project" and ".plan". To create such files, just use Pico to compose the text you want and save it under the appropriate filenames in your home directory.

To find information about a user on another system, you must know the host name of that system. At the Humanitas>

prompt, type "finger [first-name, last-name, or userid]@[host-name]". The following, for example, is the result of a finger search for Ed Krol, the author of The Whole Internet:

```
humanitas> finger krol@uxh.cso.uiuc.edu
[uxh.cso.uiuc.edu]
Login name: krol                      In real life: Ed Krol

Office: 3337886
Directory: /cso/staff/krol           Shell: /bin/csh
Last login Wed Sep  7 06:25 on ttytc from krols-pc.isdn.ui
No Plan.
```

Sometimes, other systems will report in a different format. Below, for example, is the output of a finger search for "ellison" on the humanities computer at the University of Michigan, Ann Arbor:

```
humanitas> finger ellison@um.cc.umich.edu
[um.cc.umich.edu]
Login      Name          Port      When          Office

GB9K  julie ellison          <Sep06/94 23:22> 2051 Chaucer Drive
W734  Mike Ellison          <Feb25/94 10:43> Computer Centre
W6A9  Sharon Ellison        < . . . . . > 200 Renaissance Center, Sui
W559  Ellison Remote        < . . . . . >
W3AF  Ian Ellison           <Aug15/94 16:29> Hell, Fiery Grotto #666, Lak
SXRL  John P. Ellison       <Aug19/94 12:15> tcb1240
GEAE  peter t ellison       < . . . . . >
66VR  Ian D Ellison         <Aug15/94 17:16>
```

In actuality, remote finger-searching is a hit-or-miss affair. Even if you know the name of the remote computer you wish to search (a rare occasion), that computer may not allow finger searches or will otherwise refuse to divulge its secrets.

CHAPTER 10:

LIST SERVERS ("MAILING LISTS")

10.1: THE LIST SERVER CONCEPT

One of the most valuable aspects of e-mail for humanities users is the ability to participate in (or, for the shy, simply listen in on) collective e-mail discussion groups formed around particular topics of interest. There are now several thousand such e-mail discussion groups or so-called "lists" organized by topic, discipline, period, author, etc.

Of course, if you are only communing with a small group of correspondents, it is easy to create a purely informal discussion group by setting up a list in Pine's Addressbook (see §9.7). For example, an instructor or student can create a discussion group for a course by asking everyone in the class to define a list of class members in their Addressbook under the list-title "engl236". Messages can then be broadcast to the whole class by putting "engl236" on the "To:" line. (A bit more formally: the local system administrator can set up a system "alias" or name for a preestablished group of local users.) However, any list that grows beyond a certain size or is constantly changing cannot practically be managed in this manner.

E-mail discussion groups on a national or global scale, therefore, depend on automated "list server" programs to manage their membership lists and perform other administrative tasks for them. The two most prevalent such programs for humanities scholars are "listserv" and

"majordomo". Listservs and majordomos are automated agents that sit on designated host computers somewhere with their own e-mail address--which for clarity may be called the list's "administrative address" (e.g., the administrative address for the majordomo that manages many philosophy groups is "Majordomo@world.std.com"). The idea is that users can send simple, standardized e-mail messages to these addresses to subscribe to a list, to unsubscribe, and to request certain other functions. Substantive or content-oriented messages, by contrast, go to what may be called the "list-title address" set up by the automated agent to funnel messages between individuals and the collective list membership (e.g., subscribers to the Lyotard list managed by Majordomo@world.std.com post messages to the list at large via the list-title address: "lyotard@world.std.com"). It is important to keep a clear distinction between a list's administrative and list-title addresses--since, for example, you would not want to send a subscription request to hundreds of Lyotard-fans; nor, inversely, a long analysis of the "differend" to the automatic agent.

E-mail discussion groups managed by listservs and majordomos are of two kinds: moderated and unmoderated. In a moderated list, a editor receives all postings from subscribers intended for the list-title address and decides which ones are appropriate for broadcast. Unmoderated lists, on the other hand, are like seminars without an instructor.

10.2: FINDING INTERESTING LISTS

Besides word of mouth, there are several ways to ferret out lists you might be interested in. You can get mammoth

lists of lists through Anonymous FTP, for example, at `ftp.nisc.sri.com` in the file `/netinfo/interest-groups.Z` (in compressed format). Or you can watch such Usenet newsgroups as `news.answers` for the periodic appearance of the Bitnet FAQ (Frequently Asked Question file).

However, in the case of listserv groups (which are technically part of the Internet affiliate known as Bitnet) one of the most useful methods is to send an e-mail message to any listserv with the following content: `"list global/[keyword]"`. This asks the listserv to send you an e-mail message containing a portion of the current, global list of lists. (You can also send `"list global"` to get the whole shebang, but you will receive a gigantic return message.) For example, the following message composed in Pine asks a listserv for the names of all lists with the word "literature" in the title:

To: `listserv@wvnm.wvnet.edu`
Cc:
Attchmnt:

`list global/literature`

After a few minutes (sometimes longer), you will receive two messages from the listserv. One is a short administrative message informing you that the job has been acted on (there are also some statistics regarding computing time used). The other will be the job output itself. Here, for example, is the result of the above keyword search for "literature":

From: BITNET list server at WVNVM <LISTSERV@WVNVM.WVNET.EDU>
To: Alan Liu <ayliu@HUMANITAS.UCSB.EDU>

Subject: File: "LISTSERV LISTS"

Excerpt from the LISTSERV lists known to LISTSERV@WVNVM on 8 Sep 1994 16:26
Search string: LITERATURE

Network-wide ID	Full address	List title
-----	-----	-----
AMLIT-L	AMLIT-L@MIZZOU1	American Literature Discussion Group
AUSTLIT	AUSTLIT@NDSUVM1	Austrian Literature
CDG-CNL	CDG-CNL@MIAMIU	Curriculum Development Group-Composition
CHICLE	CHICLE@UNMVMA	Chicano literature discussion list
CHILDLIT	CHILDLIT@RUTVM1	Children's Literature: Criticism and Theory
DOROTHYL	DOROTHYL@KENTVM	Mystery Literature E-conference
IAFA-L	IAFA-L@VTVM1	Scholarly discussion of Fantastic Literature
KIDLIT-L	KIDLIT-L@BINGVMB	Children and Youth Literature List
IAFA-L	IAFA-L@VTVM1	Scholarly discussion of Fantastic Literature
KIDLIT-L	KIDLIT-L@BINGVMB	Children and Youth Literature List
LITERARY	LITERARY@BITNIC	Discussions about Literature
LITSCI-L	LITSCI-L@UIUCVMD	Society for Literature and Science-philos.
MEDEVLIT	MEDEVLIT@SIUCVMB	MEDEVLIT Medieval English Literature Discu+
MENDELE	MENDELE@YALEVM	Mendele: Yiddish Literature and Language
MODBRITS	MODBRITS@KENTVM	Modern British and Irish Literature: 1895-19+
NOVELIST	NOVELIST@NERVM	"Librarians' Forum on Literature in English"
PERSIA-L	PERSIA-L@EMUVM1	Jewish Literature and History in the Persian+
PHIL-LIT	PHIL-LIT@TAMVM1.TAMU.EDU	An Electronic Symposium on Philosophy+
SEELANGS	SEELANGS@CUNYVM	SEELangs: Slavic & E. European Languages &l+
SERCITES	SERCITES@MITVMA	Citations for Serial Literature
T-AMLIT	T-AMLIT@BITNIC	Teaching the American Literatures

The left column contains list names, the middle column the beginning of the list-title address (in the Bitnet addressing convention of "[list-title]@bitnet-host"), and the right column a brief description. To subscribe or otherwise communicate with the administrative address of a list, do not use the addresses you see here (these are list-title addresses). Instead, use what you see to construct the proper administrative address as follows: "listserv@[bitnet-host].bitnet". For example, the administrative address of the CHICLE group above is "listserv@unmvma.bitnet".

- ▶ If you happen already to know the full machine name of the Bitnet machine hosting the listserv you are interested in, simply send your subscription message to "listserv@[full-address]". For example, "listserv@wvnm.wvnet.edu" (the ".wvnet.edu" part is what you need to supply in place of ".bitnet").
- ▶ Alternatively, you can send your communication to any listserv whose full machine name you know. Thus you can send a subscription request for T-AMLIT to "listserv@wvnm.wvnet.edu" and (after a long delay), it will forward your request to the appropriate machine.
- ▶ Or again, you can pass from the Internet to Bitnet through a so-called "gateway." A general-purpose

10.3: COMMUNICATING WITH THE ADMINISTRATIVE ADDRESS

Once you know the administrative address of the listserv or majordomo that manages the list you are interested in, you can send it the following requests by e-mail. (Remember that you are talking to a machine. Do not fill out the "Subject:" line in your message header; be careful not to vary from the proper message formats.) Normally, the listserv or majordomo will respond with a pair of complementary messages, one merely confirming that the job has been acted on (or informing you

there was a problem with your request), and the other containing the job output itself.

- ▶ Subscribe. To subscribe to either listserv or majordomo lists, put the following line (and nothing else) in your e-mail message: "**subscribe [list-title] [your-name]**". The following, for example, subscribes to the CHICLE group:

<p>To: listserv@unmvma.bitnet</p> <p>Cc:</p> <p>Attchmnt:</p> <p>subscribe t-amlit Alan Liu</p>

Note: a few lists ask you to send subscription requests to the human editors or moderators, who will then add you manually to the automatic list. An example is the e-mail version of Postmodern Culture, whose editors can be reached at "pmc-list@listserv.ncsu.edu".

- ▶ Unsubscribe. To unsubscribe from a listserv, put the single line in your e-mail message: "**signoff [list-title]**". Majordomos require a slight variation: "**unsubscribe [list-title]**".
- ▶ Get Information About the List. A handy feature of list servers is that you can request from them information about the rules and membership of particular lists. Send an e-mail message containing the single line "**review [list-title]**". (For majordomos, use "**who [list-title]**".) For example, the CHICLE list will mail back to you a message that begins:

<p>From: BITNET list server at UNMVMA <LISTSERV@unmvma.unm.edu></p> <p>To: ayliu@HUMANITAS.UCSB.EDU</p>

Subject: File: "CHICLE LIST"

```
*
* Chicano literature discussion list
*
* REVIEW= PUBLIC      SUBSCRIPTION= OPEN      SEND=EDITOR
* Notify= Yes          Reply-to= List,Ignore    Files= Yes
* EDITOR= tmarquez@unm.edu
* Validate= store only
* CONFIDENTIAL= NO    ACK= YES ERRORS-TO= tmarquez@unm.edu
* Notebook= Yes,L,monthly,private
*
* OWNER= tmarquez@unm.edu (Teresa Marquez)
```

zzshel@ACC.WUACC.EDU	Roy Sheldon
JELIZOND@ACCDVM.ACCD.EDU	Jorge Elizondo
JLUNALAW@ACCDVM.ACCD.EDU	Juanita Luna-Lawhn
JELIZOND@ACCDVM.ACCD.EDU	Jorge Elizondo
JLUNALAW@ACCDVM.ACCD.EDU	Juanita Luna-Lawhn
MARUBAYA@ADMINSERVER.ADM.CSUS.EDU	Kris Marubayashi
FLORES@ADMN1.LAW.UTAH.EDU	Robert L. Flores
CarterSan@AOL.COM	Kristi Carter-Sanborn
JazzyDi@AOL.COM	Diana Dominguez

This is an excellent way to collect e-mail addresses of scholars in your area of interest.

- ▶ Retrieving Files. Some listservs and majordomos cache material in files that you can download onto Humanitas selectively. If this is the case with your list, the list will usually send you the specifics (e.g., the Postmodern Culture listserv at "listserv@listserv.ncsu.edu" distributes the magazine's contents in this way). In general, you initiate a file request with listservs by sending an e-mail message with the single line: "get [filename] [filetype]" or (depending on how the remote machine is set up) "get [list-title] [filename] [filetype]". Majordomo machines require the single line "get [list-title] [filename]".

- ▶ Temporarily Stopping Mail. To stop the mail from a listserv temporarily without unsubscribing (e.g., while you are on vacation), send an e-mail message with the single line: "**set [list-title] no mail**". To start up the mail again, send the message: "**set [list-title] mail**". (Majordomos, unfortunately, do not have this option.)
- ▶ Seeing Copies of Your Own Postings: Most listservs do not send a copy of your own posting back to you when you post to the group at large. If you want to see your own messages as they are broadcast, send an e-mail message with the single line: "**set [list-title] repro**".

10.4: COMMUNICATING WITH THE LIST-TITLE ADDRESS

When you receive a posting from a list server intended for the group as a whole, it will have a header in which the "To:" line designates the list-title address. In addition the list-title address appears on the "Reply to:" line:

From: Richard Cochrane <SENRC@CARDIFF.AC.UK>
 Reply to: lyotard@world.std.com
 To: lyotard@world.std.com
 Subject: First extract: some thoughts

- ▶ Posting a Message on a New Subject. To post a message to the list on a new topic, just put the list-title address on the "To:" line in your own message (e.g., "lyotard@world.std.com). Alternatively, reply to an

existing message and simply edit the "Subject:" line in the resulting header.

- ▶ Replying to a Message. When you use the "r" for reply option in Pine to respond to a list message, you'll have a choice to make between replying just to the individual who originated the message (Richard Cochrane on the "From:" line above) or to the list at large ("lyotard@world.std.com" on the "Reply to:" line). Pico will ask, "Use 'Reply to:' address instead of 'From:' address? (y/n/^C)". Answer "y" or "n." Pine next prompts you to fill in the "Cc:" line by asking, "Reply to all recipients? (y/n/^C)". If you already responded "yes" to the previous question regarding the "Reply to" address, answer "no" here, since otherwise the list-title address will appear redundantly on the "Cc:" line.

Note: some list servers are smart enough to protect users against the common mistake of sending a subscription or other such request to the list-title address instead of the administrative address. After you've sent off your misaddressed request, you'll receive e-mail from the list server machine saying that it has intercepted your posting because it "looks like" it was intended for the other address.

☛ E-mail list servers are not the only medium of collective, group discussion or file caching on the Internet. An alternative medium is the Usenet, whose newsgroups are similar to e-mail discussion groups (indeed, many e-mail lists lead a double life on Usenet under the .bit.listserv hierarchy). As regards file caches or archives, Anonymous FTP, Gopher, and World Wide Web also offer means to retrieve files (some entities such as Postmodern Culture allow users to download material through several interfaces).

CHAPTER 11:

FTP (AND ARCHIE & WAIS SEARCH ENGINES)

11.1: THE FTP CONCEPT

"FTP" or "File Transfer Protocol" is a program (and protocol) that allows users to logon to remote computers in a passive, read-only mode and retrieve archived files. (It is also a means of handling purely local file transfer tasks; see §2.2-2.3.) Though it is one of the oldest and most restrictive interfaces on the Internet, FTP shares with e-mail the distinction of being a staple of Net life. This is because while it is terrible at some tasks (browsing files through the FTP interface is about as efficient as harvesting grain with a knife), it is singularly adept at the one thing it does well: zeroing in on an exact file anywhere in the world and whisking it back home (to Humanitas).

The particular subset of FTP services that you will most often use is "Anonymous FTP." Anonymous FTP allows you to be a guest on a remote computer that someone has generously set up as an FTP-server with file archives open to the public. You logon to the FTP-server under the userid "anonymous" and an optional password (by convention: your e-mail address). If the FTP-server is not overloaded (it will deny access when there are too many outsiders), you will be admitted to a public area in the host system's directory tree and allowed to exercise a limited range of commands (similar to Unix commands). You can move from the directory you are initially placed in down

through its subdirectories and sub-subdirectories and back again, but not elsewhere in the system's tree; and you can "read" but not write or edit (though some systems allow you to upload your contributions to a special, quarantine subdirectory named something like "incoming"). Essentially, you shop: you see and order things "as is."

11.2: ANONYMOUS FTP LOGON/LOGOFF

To logon by Anonymous FTP on a remote FTP-server, go "ftp [FTP-server-address]" at the *humanitas*> prompt. Alternatively, first go "ftp" to open the FTP program, then from the ftp> prompt go "open [FTP-server-address]".

Once the connection to the FTP-server is made, you will be asked for a userid or name. Enter "anonymous". You will next be prompted for a password. Enter your e-mail address. (Those dialing in by modem from home may want to set up a macro or script in their communications program to supply Anonymous FTP login information automatically.) If the remote system is overloaded, it will at this point deny access. The kinder systems also inform you how many anonymous users are allowed at any one time, when would be a better time to try, and whether there are any "mirror sites" with the same file archives.

Once you make it past the front door, you will usually be greeted with some information specific to the system (e.g., a suggestion that you look at a "readme" file with instructions or descriptions of resources). Then you will see an ftp> prompt indicating that the remote system is at your command.

For example, the following opens a session with the site *ftp.std.com*:

```
humanitas> ftp
ftp> open ftp.std.com
Connected to ftp.std.com.
220 ftp FTP server (Version wu-2.4(3) Thu Apr 14 16:20:44 EDT 1994) ready.
Name (ftp.std.com:ayliu): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password: ayliu@humanitas.ucsb.edu [note: password not echoed on screen]
230-
230-Hello!
230-
230-This is the anonymous FTP area for world.std.com, a public access Unix
230-system. Accounts directly on the system are available via telnet or
230-direct-dial (617-739-9753, 8N1, V.32bis (14.4K), V.32 (9600), 2400, etc.),
230-login as new (no password) to create an account. Accounts are charged
230-at $5/mo+$2/hr or $20/20hrs/month, your choice. Grab the details in
230-the world-info directory here if interested.
230-
230-
230-Please read the file README
230- it was last modified on Tue Apr 5 22:43:01 1994 - 157 days ago
230 Guest login ok, access restrictions apply.
ftp>
```

To logoff from the remote system and return to the *humanitas>* prompt, go *"quit"*. (*"quit"* exits both the FTP program running on the remote server and the local-client FTP program running on Humanitas; if you were sitting at the *ftp>* prompt in the local client without a remote connection active, *"quit"* simply exits the local client).

If for some reason you get *"stuck"* somewhere in the remote FTP-server and can't move even to break the connection and get back to Humanitas, try the usual Unix escape measures first (*"^C, quit, bye, stop"* etc.). As a last

resort, hit the "break" key on your keyboard, which dumps everything and makes you login to Humanitas again.


11.3: BROWSING REMOTE FTP-SERVERS

Once you are sitting at the `ftp>` prompt on a remote FTP-server, you can move around the public-access directories, list directory contents, and browse files by using a handful of commands similar to Unix commands:

- ▶ **"dir"** lists the contents of your current directory in a format like the `"ls -l"` command in Unix. This is usually the best way to see what is in a directory. On some systems, the use of a wildcard (`"dir *"`) lists contents recursively such that you see not only the contents of the present directory but also the contents of any subdirectories (this is the same as the command, `"ls -lR"`).

If the directory listing is so long that most of it rolls past your screen before you can read it, use `"dir * "|pg"`. (Note the quote marks around `"|pg"`; `"|"` is the so-called "pipe" symbol in Unix. Note also that this only works if you use the wildcard `"*"`. If you omit the `"*"`, the effect will be to dump the output of the `dir` command to an actual Unix file in your space on Humanitas.) This will pipe the output of the `dir` command to the `pg` reading program on Humanitas, which displays it a screen at a time. Alternatively, you can use `more` instead of `pg`. (Users dialing in from home can also use the "screen review" feature in their communications program to see what has streamed by.)

If you find yourself trapped in a very long listing that continues to scroll on, and on, and on, go "**^C**" to return to the ftp> prompt.

 Some tips on interpreting FTP directory listings: a "d" at the beginning of the permissions information (e.g., "drwxrwxr-x") means that the name at the right of the screen (e.g., "obi") is a subdirectory. A blank at the beginning of the permissions information (e.g., "-rwxrwxr-x") means that the name at the right is a file. If you see a listing that looks like the following, it means that "/customers/periodicals" is the actual subdirectory but "periodicals" "points" to it. You can use "**cd periodicals**" to move to that directory:

Here, "2511"= 2.5K. This is an especially important piece of information to check before starting any transfer operations.

🔍 Look for files called "readme", "index.txt", or "OOindex.txt". They often contain descriptions of archive contents.

- ▶ "ls" is similar to *dir* but produces a simplified listing with only the names of subdirectories and files (and no way to tell the difference). This is most useful when you have already navigated to a bottom-level subdirectory holding only files. To read the output of ls a screen at a time, use "ls * "|pg"". ("ls -l" is the same as "dir"; "ls -lR" is the same as "dir *". Note that unlike "dir *", "ls *" does not produce a recursive listing.)
- ▶ "cd [directory-name]" moves you to another directory in the public-access tree. If you are changing to a subdirectory of your current working directory, you can simply type the target directory name without a slash (e.g., "cd obi"). But you can also jump several directory levels by using path names (e.g., "cd /obi/John.Keats").

- ▶ *"cdup"* moves you back up the tree one level. Note that this is divergent from Unix commands. There is no *cdup* command in Unix; on the other hand, while FTP has *cdup* it doesn't have Unix's convenient *"cd"* (used by itself) for jumping back from wherever you are to your original home directory.
- ▶ *"pwd"* tells you what directory you are presently in (in case you get lost).
- ▶ *"get [filename] |pg"* displays a file a screen at a time using the *pg* program. (Alternatively, you could use *more*.) Since this requires that the file on the remote server be shipped home to Humanitas and finally to your screen, there will be a delay. In the case of short files, the delay is usually negligible (though it can be longer during busy times of the day when the Internet is clogged with traffic). But longer files can take a minute or two. And very long files can take a while. Be sure to check the file size in the directory listing before "getting" a file. File size is reported in bytes (e.g., 13,102 bytes=13K). Any file in the six-figure byte range and above (100K or more) will be slow in the getting. Do not "get" files named something like *"paradise.lost"* or *"shakespeare.tragedies"* unless you are prepared to wait a LONG TIME to receive a BIG TEXT. If you find yourself trapped in a "get" operation and want to interrupt it, *"^C"* usually stops the transfer and returns you to the *ftp>* prompt. (Because of the vagaries of the Internet, occasionally even short transfers will get hung up somewhere between Humanitas and the remote FTP-server. Trying again a few minutes later often clears up the problem.)

The following is a sample FTP session in which, after having already logged onto the ftp.std.com server (as per above), I navigate through the Online Books Initiative directories, list their contents in various ways, and then bring home for display some poems of Keats (I silently elide long listings to save space; comments are added in italics):

```
ftp> dir      [display contents of current directory]
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/lis.
total 394
-rw-r--r--   1 0   daemon  145157 May 28  1992 .find.codes
drwx-----   2 114 daemon    512 Mar 15  1993 .obs
lrwxrwxrwx   1 0   src      18 Apr  6 01:02 AW -> customers/books/AW
lrwxrwxrwx   1 0   src      22 Apr  6 00:58 Kluwer -> customers/books/Kluwer
drwxr-xr-x   3 0   daemon   512 Feb 11  1994 OBS
lrwxrwxrwx   1 0   src      23 Apr  6 00:52 Quantum ->customers/books/Quantum
-rw-rw-r--   1 0   src      2511 Apr  6 02:43 README
lrwxrwxrwx   1 0   src      23 Apr  6 01:05 Softpro ->customers/books/Softpro
drwxrwxr-x   7 108 10      512 Aug  3 19:35 WWW
lrwxrwxrwx   1 0   src      24 Apr 17 12:56 community->/ftp/customer/community
lrwxrwxrwx   1 0   src      21 Apr  6 01:12 consultants->customers/consultants
drwxrwxr-x 215 0   src      5120 Jul  4 16:07 obi
drwxrwx--x   3 0   10      512 Jun 29 22:17 private
drwxrwxrwt 183 0   0        5632 Sep  8 22:24 pub
drwxrwxr-x  71 0   10      1536 Aug 31 16:53 src
lrwxrwxrwx   1 0   src      17 Apr  7 01:06 vendors -> customers/vendors
drwxr-xr-x   2 0   src      512 Mar 22 16:50 world-info
226 Transfer complete.
2302 bytes received in 1.8 seconds (1.2 Kbytes/s)

ftp> cd obi   [change to "obi" subdirectory]

ftp> pwd      [check what directory I'm in]
257 "/obi" is current directory.

ftp> dir      [list directory contents]
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/lis.
total 6662
drwxrwxr-x   2 108 10      1536 Sep 30  1992 .cap
-rw-rw----   1 552 552      0 Jun 29 21:37 .forward
-rw-rw-r--   1 990 src      637 Oct  6  1990 00-README
```

```

drwxrwxr-x  2 990  src      512 Sep 30  1992 A.E.Housman
drwxrwxr-x  2 990  src      512 Sep 19  1991 A.Hofmann
drwxrwxr-x  2 990  src      512 Dec 30  1992 ACN
drwxrwxr-x  2 990  src     1536 Sep 19  1991 ATI
drwxrwxr-x  2 990  src      512 Sep 19  1991 Access
drwxrwxr-x  2 990  src      512 Dec 20  1991 Aesop
drwxr-xr-x  2 990  src     1024 Nov  7  1993 Algernon.Charles.Swinburne
drwxr-xr-x  2 990  src      512 Mar 22 21:13 Ambrose.Bierce
drwxrwxr-x  2 990  src      512 Sep 19  1991 Amoeba
drwxr-xr-x  3 990  src      512 Dec  4  1992 Anarchist
drwxr-xr-x  2 102  10      512 Feb 16  1994 Andrew.Marvell
drwxrwxr-x  4 990  src      512 Jun 22 00:02 Anglo-Saxon
drwxr-xr-x  2 990  src      512 Nov 19  1991 Anonymous
drwxr-xr-x  2 990  src      512 Apr 12  1993 Ansax
drwxr-xr-x  2 990  src      512 Oct  1  1992 Antarctica
drwxrwxr-x  2 990  src      512 Sep 19  1991 ArtCom
drwxrwxr-x  7 990  src     1024 Jan 25  1994 Arthur.Conan.Doyle
drwxrwxr-x  2 990  src      512 Sep 19  1991 Athene
drwxrwxr-x  3 990  src      512 Jul 19  1993 BSD

```

^C [Interrupt operation because the listing turns out
to be long and is just streaming by on the screen.]

426 Transfer aborted. Data connection closed.

226 Abort successful

local: |pg remote: *

23874 bytes received in 1.1e+01 seconds (0.36 Kbytes/s)

ftp> dir *"|pg" [display directory contents recursively a screen at a time in pg]

200 PORT command successful.

150 Opening ASCII mode data connection for /bin/ls.

```

-rw-rw-r--  1 990  src      637 Oct  6  1990 OO-README
lrwxrwxrwx  1 0    src       9 Mar 22 21:16 Eduard.Douwes.Dekker -> Multatuli
-rw-rw----  1 990  src     2280174 Mar 27  1992 Shakespeare.complete.Z
-rw-rw-r--  1 990  src      862 Mar 27  1990 Titles
lrwxrwxrwx  1 0    0        6 Mar 16 03:32 Usenix -> USENIX
-rw-rw-r--  1 990  src       0 Jan 23  1991 desert-storm.irc.5
-rw-r--r--  1 990  src     3506 Feb  1  1993 tahiti

```

A.E.Housman:

total 8

```

-rwxr-xr-x  1 0    daemon    59 Sep 30  1992 .cache
-rw-rw-r--  1 990  src     2837 Sep  4  1990 Terance

```

John.Donne:

total 4

```

-rw-r--r--  1 102  10     813 Feb 16  1994 Song
-rw-r--r--  1 102  10     277 Feb 16  1994 Song~

```

:<Return> *[Go to the next screen in the pg display]*

John.Keats:

total 164

```
-rw-r--r-- 1 990 src 704 Oct 1 1993 devon.maid
-rw-r--r-- 1 990 src 18109 Oct 1 1993 eve.st.agnes
-rw-r--r-- 1 990 src 683 Oct 1 1993 hither.love
-rw-r--r-- 1 990 src 1228 Oct 1 1993 hymn.to.apollo
-rw-r--r-- 1 990 src 37803 Oct 1 1993 hyperion
-rw-r--r-- 1 990 src 688 Oct 1 1993 in.drear.nighted.december
-rw-r--r-- 1 990 src 1681 Oct 1 1993 la.belle.dame
-rw-r--r-- 1 990 src 669 Oct 1 1993 lines
-rw-r--r-- 1 990 src 1041 Oct 1 1993 meg.merrilies
-rw-r--r-- 1 990 src 847 Oct 1 1993 o.blush.not.so
-rw-r--r-- 1 990 src 2846 Oct 1 1993 ode.on.indolence
-rw-r--r-- 1 990 src 3767 Oct 1 1993 ode.to.a.nightingale
-rw-r--r-- 1 990 src 1629 Oct 1 1993 ode.to.autumn
-rw-r--r-- 1 990 src 1904 Oct 1 1993 robin.hood
-rw-r--r-- 1 990 src 641 Oct 1 1993 think.of.it.not
-rw-r--r-- 1 990 src 2390 Oct 1 1993 to.hope
-rw-r--r-- 1 990 src 645 Oct 1 1993 to.sleep
-rw-r--r-- 1 990 src 0 Oct 1 1993 you.say.you.love
```

John.Milton:

total 416

```
-rw-r--r-- 1 102 10 203785 Oct 28 1991 Paradise.Lost.Z
```

^C *[interrupt operation (the listing is very long and I've
already found something I want to see)]*

426 Transfer aborted. Data connection closed.

226 Abort successful

local: |pg remote: *

46182 bytes received in 1.2e+02 seconds (0.37 Kbytes/s)

ftp> cd John.Keats *[move from "obi" dirctory to "John.Keats" subdirectory]*

250 CWD command successful.

ftp> dir

200 PORT command successful.

150 Opening ASCII mode data connection for /bin/ls.

total 164

```
-rw-r--r-- 1 990 src 704 Oct 1 1993 devon.maid
-rw-r--r-- 1 990 src 18109 Oct 1 1993 eve.st.agnes
-rw-r--r-- 1 990 src 683 Oct 1 1993 hither.love
-rw-r--r-- 1 990 src 1228 Oct 1 1993 hymn.to.apollo
-rw-r--r-- 1 990 src 37803 Oct 1 1993 hyperion
-rw-r--r-- 1 990 src 688 Oct 1 1993 in.drear.nighted.december
```

```
-rw-r--r-- 1 990 src 1681 Oct 1 1993 la.belle.dame
-rw-r--r-- 1 990 src 669 Oct 1 1993 lines
-rw-r--r-- 1 990 src 1041 Oct 1 1993 meg.merrilies
-rw-r--r-- 1 990 src 847 Oct 1 1993 o.blush.not.so
-rw-r--r-- 1 990 src 2846 Oct 1 1993 ode.on.indolence
-rw-r--r-- 1 990 src 3767 Oct 1 1993 ode.to.a.nightingale
-rw-r--r-- 1 990 src 1629 Oct 1 1993 ode.to.autumn
-rw-r--r-- 1 990 src 1904 Oct 1 1993 robin.hood
-rw-r--r-- 1 990 src 641 Oct 1 1993 think.of.it.not
-rw-r--r-- 1 990 src 2390 Oct 1 1993 to.hope
-rw-r--r-- 1 990 src 645 Oct 1 1993 to.sleep
-rw-r--r-- 1 990 src 0 Oct 1 1993 you.say.you.love
```

226 Transfer complete.

1051 bytes received in 2.9 seconds (0.36 Kbytes/s)

ftp> get la.belle.dame |pg [display file in pg]

200 PORT command successful.

150 Opening ASCII mode data connection for la.belle.dame (1681 bytes).

LA BELLE DAME SANS MERCI

John Keats

Ah, what can ail thee, wretched wight,

Alone and palely loitering?

The sedge is withered from the lake,

And no birds sing.

Ah, what can ail thee, wretched wight,

So haggard and so woe-begone

The squirrel's granary is full,

And the harvest's done.

I see a lily on thy brow

With anguish moist and fever dew,

And on thy cheek a fading rose

Fast withereth too.

I met a lady in the meads,

Full beautiful, a faery's child:

Her hair was long, her foot was ligh,

And her eyes were wild.

:

I set her on my pacing steed,

And nothing else saw all day long;

For sideways would she lean, and sing

A faery's song.

I made a garland for her head,

And bracelets too, and fragrant zone;
 She looked at me as she did love,
 And made sweet moan.

She found me roots of relish sweet,
 And honey wild, and manna dew,
 And sure in language strange she said,
 "I love thee true!"

She took me to her elfin grot,
 And there she gazed and sighed deep,
 And there I shut her wild, sad eyes---
 So kissed to sleep.

And there we slumbered on the moss,
 And there I dreamed, ah! woe betide,
 :

The latest dream I ever dreamed
 On the cold hill side.

I saw pale kings, and princes too,
 Pale warriors, death-pale were they all;
 Who cried---"La belle Dame sans merci
 Hath thee in thrall!"

I saw their starved lips in the gloam,
 With horrid warning gaped wide,
 And I awoke and found me here,
 On the cold hill side.

And that is why I sojourn here,
 Alone and palely loitering,
 Though the sedge is withered from the lake,
 And no birds sing.

(EOF): <Return>

226 Transfer complete.

local: |pg remote: la.belle.dame

1745 bytes received in 9.2 seconds (0.19 Kbytes/s)

ftp> quit[close remote connection; exit FTP; return to humanitas> prompt]

221 Goodbye.

humanitas>

11.4: DOWNLOADING FILES FROM REMOTE FTP-SERVERS

Once you have found a file or files you want to download from an FTP-server, you can ship it back home to Humanitas to a Unix file in your storage space.

A. Text Files: If the file you want is a straight, uncompressed text file, use either of these two commands:

- `get [remote-file-name]`. For example, if you want to transfer the file "la.belle.dame" in the John.Keats directory above, you would change to the directory and go "`get la.belle.dame`". The FTP program will not display the file as it is being transferred, but it will report that the operation has begun, the number of bytes transferred, and the amount of time it took. It will then return you to the ftp> prompt. The file at this point will have been transferred under its original name to your home directory on Humanitas. If you want to transfer the file and save it under a different name, add a local name as follows: "`get [remote-file-name] [local-file-name]`". The local name can include a path.
- `mget [remote-file-name-1] [remote-file-name-2] [etc.]`. This command allows you to transfer multiple files, though FTP will ask you to confirm with a "y" before each one. Files will be placed

under their original names in your home directory. To force **mget** to transfer all specified files without asking for confirmations, first issue the command, "**prompt.**" This toggles off the confirmation feature; issuing "**prompt**" once more toggles it on again.

When the file has arrived home to Humanitas, you can then manipulate it as you wish in Unix or download it to your local pc (see §2.1-2.4).

B. Binary Files: If the file you want is a binary file (a program, graphic, audio, or any kind of compressed or archived file [with such extensions in the name as ".z", ".zip", ".gz", or ".tar"], you need to take one preliminary step.

- ▶ While at the ftp> prompt (and connected to the remote FTP-server), go "**binary**". This toggles FTP into the binary transfer mode (which can usually also transfer straight text or ASCII files). FTP will remain in binary transfer mode during the current session until you toggle back by going "**ascii**". By default, FTP opens in ascii transfer mode.
- ▶ Once you have set the mode to binary, then use **get** and **mget** as per above.

In the following example, I have already logged onto the FTP-server at ftp.ncsa.uiuc.edu and have moved down several levels in the tree to the directory whose full path is /Mosaic/Contrib/Windows/viewers. While in this "viewers" directory, I download a graphics-viewing program called LiveView:

```

ftp> binary
200 Type set to I.
ftp> get lview31.zip
200 PORT command successful.
150 Opening BINARY mode data connection for lview31.zip (224269 bytes).
226 Transfer complete.
local: lview31.zip remote: lview31.zip
224269 bytes received in 13 seconds (16 Kbytes/s)
ftp> quit
221 Goodbye.
humanitas>

```

🐼 Let's say that you want to download a gigantic file from a remote FTP-server (e.g., a program) to Humanitas in order subsequently to download it from Humanitas to your home computer. But your allowed storage space on Humanitas is too small. How do you get around this bottleneck? The solution is to use the `/tmp` directory on Humanitas as a temporary holding pen. While in a remote FTP session, go `"get [filename] /tmp/[filename]"` to download the file to `/tmp`. (While `/tmp` is a public access directory, individual files are created in it with owner's permission only; no one else, except the system administrator, can read it.) Later, you can ship the file from `/tmp` to your local or home computer.

11.5: ISSUING LOCAL COMMANDS WHILE IN A REMOTE FTP SESSION

Sometimes you need to be able to access your storage space on Humanitas while in the middle of a remote FTP session. For example, you've accidentally downloaded some huge files from the remote FTP-server and you want to erase them before continuing. You want to rename a file you have

just downloaded called "readme" because you now want to download a different file also called "readme". Or again, you want to check how much space you still have left.

You can issue a limited subset of local Unix commands to Humanitas while at the ftp> prompt of a remote session by prefacing the command with an exclamation point: "!" For example, "!ls" lists the files in your current working directory on Humanitas (by contrast, "ls" would list files in your current working on the remote FTP-server). Most file management commands also work (be careful: the "rm" command in this context does not prompt for confirmation). However, you cannot change your working directory in Humanitas to a new directory (though you can list the contents of other directories by going "!ls [directory]"). Nor do any aliases you have set up in your .cshrc file work.

When you are done with the local operation, FTP returns you to the ftp> prompt in your remote session.

11.6: ARCHIE AND WAIS SEARCH ENGINES

The above discussion of Anonymous FTP has assumed either that you already know what remote FTP-server (and what directory on that server) to go to for the materials you seek or that you are in a mood to browse servers and directories serendipitously. But sometimes, of course, you won't know where your object of desire is and don't want to wander at random. The solution is to use a search engine.

There are two search engines for Internet file archives: Archie and WAIS. Archie is the most dependable and

comprehensive, but is limited to searches for file names. By contrast, WAIS (an acronym for "Wide Area Information Service") is quirky and selective but has the unique ability to search vast amounts of file contents. (Think of WAIS as a sort of mad librarian who knows everything there is to know about just a few, random floors of the library and--when actually working as opposed to sleeping--is liable to answer a request for material on "English literature" with references to the entirety of texts written in English.)

ARCHIE

There are ten "Archie servers" in the world known to the Archie local-client program on Humanitas (there are actually more such servers). An Archie server is a computer that periodically and systematically connects to every known FTP-server in the world and takes a "snapshot" of its file listings. When you connect with an Archie server, you can search those listings to learn which FTP-servers hold the file you are looking for. The ten Archie servers known to Humanitas are:

- archie.ans.net (USA [NY])
- archie.rutgers.edu (USA [NJ])
- archie.sura.net (USA [MD])
- archie.unl.edu (USA [NE])
- archie.mcgill.ca (Canada)
- archie.funet.fi (Finland/Mainland Europe)
- archie.au (Australia)
- archie.doc.ic.ac.uk (Great Britain/Ireland)
- archie.wide.ad.jp (Japan)

archie.ncu.edu.tw (Taiwan)

By default, Archie at Humanitas currently connects to the archie.sura.net server. If archie.sura.net (or an intervening link) is down when you need it, you can use the "-h" option in the "archie" command (described below) to tap a different Archie server.

To use the Archie client on Humanitas, type the following at the humanitas> prompt: "archie [-options] [keyword]". If you include no options, Archie searches by default for an exact, case-sensitive filename match and returns up to 95 matches. The output will include the name of the remote computer (the "host"), the directory on that computer where the file can be found, the full filename, and the file size and creation date. If you know the exact filename you are looking for, this is the most efficient way to use Archie.

However, you will often want to take advantage of some of Archie's other search options. The most useful options (which can be combined) are the following:

- ▶ -s "archie -s [keyword]" initiates a case-insensitive substring search. "Substring" means that matches will include any filename that includes your keyword. For example, an exact search for "view" will yield only files named "view" (as well as "view.zip", "view.gz", etc.; the extension doesn't figure in the equation). But a substring search for "view" will also yield "lview", "lview31.zip", "overview", "ghostview", etc. Because the -s option is case insensitive, such a search would also find "LVIEW", "Lview", "lVIEW", and so on. (Remember that

most of the Internet and Unix universe is by default case-sensitive).

- ▶ **-c** `"archie -c [keyword]"` initiates a case-sensitive substring search. This works just like the **-s** option described above, but is case-sensitive.

- ▶ **-r** `"archie -r [keyword]"` initiates a "regular expression" substring search. "Regular expression" refers to special characters that Unix knows how to use as wildcards. The subject of regular expressions is too complicated to go into here. A few useful options are as follows:
 - A keyword expressed as `"l.view"` will find `"lview"`, `"smallview"`, `"little_view"`, etc. That is, the period (`"."`) stands for any character or characters.

 - A keyword expressed as `"^view"` finds `"view"`, `"viewfinder"`, `"viewers"`, etc. The carat (`"^"`) means that any filename found must begin with the specified substring.

 - A keyword expressed as `"view$"` finds `"view"`, `"lview"`, `"ghostview"`, etc. The dollar sign (`"$"`) means that any filename found must end with the specified substring.

- ▶ **-m[n]** Typing `"archie -m[n] [keyword]"` (where `"n"` is a number) causes the search to yield no more than the first `"n"` matches. By default, Archie will harvest as many as 95 matches. This is not a problem if the

file you are seeking is relatively rare on the Net. But a substring search for "zip", for example (the file compression/decompression programs zip, pkzip, gunzip, etc. are among the most common utilities on the Net), would produce 95 matches without even burping. Typing `"archie -m10 -s [keyword]"` yields a more manageable harvest. Caution: in the case of common substrings, setting "n" too low may yield an unrepresentative sampling of the overall pool of filenames. In the case of "zip," for example, a low "n" will produce only files named "gzip" for no other reason than the fact that this is what Archie first finds. To see all the varieties of files with "zip" somewhere in the title, you would need to set "n" to an extremely high number (e.g., 500). When you find yourself in such a bind, it usually means that your choice of a keyword or search option is ill-conceived. Try a keyword that is likely to be more discriminating in what it finds. Or try a -r search with a "^" or "\$" operator.

- ▶ `-t` `"archie -t [keyword]"` initiates a search (an exact, case-sensitive search, unless you also add one of the other options described above) whose output will be listed in order of time of file creation (in descending order from latest to earliest). This is useful because there are often many versions of a program or text file on the Net simultaneously. The `-t` option groups the most recent versions at the top.

- `-h` "`archie -h [archie-server-name] [keyword]`" initiates an Archie search by way of an Archie server other than the default (see above). For all practical purposes, Archie servers are equivalent. You would only need to change servers if the default is not working at present. To conserve bandwidth on the Net, pick an alternate server as close to home as possible.

The following is a sample Archie session featuring a case-insensitive, substring search for "pkzip" with a maximum of 5 hits:

```
humanitas> archie -m5 -s pkzip

Host aix1.segi.ulg.ac.be

    Location: /local/msdos
    FILE -rw-r--r--      202574  Dec 16 1993  pkzips.exe

Host hermes.hrz.uni-bielefeld.de

    Location: /.mntO/systems/win3x
    FILE -rw-r--r--      42166  Feb 15 00:00  pkzip.exe

Host ajpo.sei.cmu.edu

    Location: /public/piwg/piwg_11_92/piwg
    FILE -rw-r--r--      31342  Jan 21 1991  pkzip.exe

Host uceng.uc.edu

    Location: /pub/wuarchive/languages/ada/ajpo/piwg/piwg_11_92/piwg
    FILE -r--r--r--      31342  Jan 20 1991  pkzip.exe

Host uvaarpa.virginia.edu

    Location: /pub/pc
    FILE -rw-rw-rw-      32880  Aug  3 1993  pkzip.exe
```

The output of this search tells me, for example, that I can Anonymous FTP to "uvaarpa.virginia.edu" (the FTP-server or "host"), use the "cd" command to move to the directory

`"/pub/pc"`, and then `"get"` the file named `"pkzip.exe"`. The file is a relatively small 33K in size.

WAIS

WAIS has at its command about 500 indexed databases of file contents, each organized around some particular topic (e.g., `"bryn-mawr-clasical-review [sic]"`, `"bush-speeches,"` `"Queer-Resources"`). If you find a database relevant to your interests and your WAIS client is able to connect to it (the databases are distributed on WAIS servers throughout the Net), you can search for one or more keywords in files, view the files you find, and then e-mail texts back to yourself. However, in my experience there are currently few databases relevant to the humanities; and half the time WAIS cannot make the proper connection. *Caveat emptor.*

Humanitas currently does not have a local-client WAIS program, so users are restricted to `"telneting"` to remote, public-access WAIS clients at a handful of sites (i.e., computers set up to offer a limited subset of WAIS client-services to anyone). This adds to WAIS's innate quirkiness an extra level of counter-intuition. As the oxymoron `"remote local client"` might indicate, working on a long distance WAIS client is as awkward as trying to order clothes to fit from a mail-order company.

These caveats aside, however, accomplished WAIS users swear that there is an `"art,"` or possibly zen, of WAIS. The trick is to ask the oracle the right question in the right way. (As it were: `"What has four legs in the morning, two at noon, and three at night?"`)

There are seven or so telnet sites for public-access WAIS clients, but at the time of this manual's writing only two

actually allowed outsiders in, worked as promised, and had access to enough of the total number of databases to warrant exploration. These two sites are:

info.funet.fi (Finland)
quake.think.com (Massachusetts, USA)

(Others include: nnsf.nsf.net, kudzu.cnidr.org, and sunsite.unc.edu; also there is swais.cwis.uci.edu, the UC Irvine WAIS client limited to UC Irvine concerns.) The Finland site presents you with all 500 databases (approximately), from which you "select" the ones to use for your search. The Massachusetts site is organized in a newer style adapted to the growing number of databases. It first presents you with a single "directory of servers" or database of databases that you search by keyword to locate relevant databases among the total 500. Then, in a second stage, you search the databases you've found.

To begin a WAIS search, go "telnet info.funet.fi" or "telnet quake.think.com" at the `humanitas>` prompt. This will connect you by telnet to the appropriate public-access WAIS clients. At the login prompt, type "wais". Then proceed as follows depending on the machine:

info.funet.fi

When prompted by the Finland machine, answer successively that you want the "wais" service, the "swais" interface (swais is "simple" or line-mode WAIS), and the "vt100" terminal type.

You will then be presented with the first screen in a menu of approximately 500 databases or "sources." Each item in the

menu is prefixed by an ordinal reference number (e.g., database number "8") and shows the title of the database. Move the highlight bar through the list as follows:

- **Up, Down Arrow Keys** move an item at a time.
- **"J"** (upper-case J) moves to the next screen of the list.
- **"K"** (upper-case K) moves to the previous screen of the list.

Since the titles of databases can be cryptic, you may wish to view a fuller "description":

- **"v"** calls up the description of the item under the highlight bar (v uses the **more** pager)

To select a database or source to use in your search:

- **<Spc>** "selects" the database under the highlight bar for use in your current session. You'll see an asterisk next to the item ("*").

Once you have finished selecting one or more databases to use, then initiate a search as follows:

- **<Return>** calls up a "Keywords:" prompt at the bottom of the screen. Enter one or more keywords separated by spaces. Then press **<Return>** once more to begin the search. If you make a mistake typing the keywords, you cannot delete or

backspace. Instead you must go "**^U**" to clear the entry and start over.

- WAIS at this point will "initialize connection" with the machine(s) on which the databases you selected reside (if a connection fails, WAIS goes on to the next selected database; if you get hung up here, **^C** out). It will then quickly search the file contents indexed in those databases for your keywords. If there is a match on one or more keywords, it will present you with a menu of descriptive titles and file sizes. In addition, it will report a "score" for each found item on a relative scale of 1000. A score of "1000" means the item is the best match found. (WAIS, unfortunately, has a very limited notion of what constitutes a good match. Since it is not context-sensitive, it thinks that the number of keyword matches per document is the same as quality of match.)
- If you see a file of interest, position the highlight bar over it and press **<Spc>**. WAIS will take care of making the appropriate machine connections and then display the file on screen. "**q**" exits the document and returns to the menu of found items. If you decide you want a copy of the file, finish viewing the document or "**q**" out of it to return to the menu of found items. Then go "**m**" (and fill in your e-mail address when asked) to have the file mailed back to yourself.

To do a new search, follow this procedure:

- ▶ *"s" Pressing "s" while looking at the menu of found items in a previous search returns you to the list of sources (i.e., the 500 original databases).*
- ▶ *"=" clears the selections you previously made among those databases (marked by asterisks). Or you can just stay with the same databases.*
- ▶ *<Return> or "w" for "words" puts you back in the "Keywords:" prompt with your previous keywords still displayed. Go "^U" to clear the entry and type in new keywords. Then <Return> to start your search.*

You can quit WAIS as follows:

- ▶ *"q" closes the connection with the remote WAIS client and returns you to the humanitas> prompt.*
- ▶ *If you get stuck somewhere while using a remote WAIS client, use the standard telnet escape key, "^]", to reach the telnet> prompt. Go "quit" to get back to humanitas>. (On Telnet, see §12.1-12.2.)*

Here is the beginning of a WAIS session with the Finland machine:

```
telnet info.funet.fi
Trying 128.214.6.102 ...
Connected to archie.funet.fi.
Escape character is '^]'.

SunOS UNIX (archie.funet.fi)
```

Finnish University and Research Network FUNET
Information Service

The following information services are available:

gopher	Menu-based global information tool
www	World Wide Web, Global hypertext web
wais	Wide Area Information Server, global databases on on different topics
x500	X.500 clients are on nic.funet.fi, login: dua, no password
archie	Database of Internet Archive contents
exit	Exit FUNET information services

**** Do NOT attempt to MAIL to yourself from Lynx et.al., unless you ****
**** are absolutely positive that the address you enter for yourself ****
**** really is globally workable INTERNET address! Any other kind ****
**** of addresses are likely to fail without any message sent to you ****
**** Specifically we do NOT support any pseudo-domains, like BITNET, ****
**** and UUCP.. ****

Select service (gopher/www/wais/archie/exit) ? **wais**

Select WAIS interface:

swais VT100-based WAIS client

Select interface (return for back to main menu) ? **swais**

Supported terminal types are:

vt100, xterm

Please enter your terminal type (vt100) ? **vt100**

Starting WAIS ..

SWAISSource SelectionSources: 475

#ServerSourceCost

001:	[archie.au]	aarnet-resource-guideFree
002:	[munin.ub2.lu.se]	academic_email_confFree
003:	[wraith.cs.uow.edu.au]	acronymsFree
004:	[archive.orst.edu]	aeronauticsFree
005:	[ftp.cs.colorado.edu]	aftp-cs-colorado-eduFree
006:	[nostromo.oes.orst.ed]	agricultural-market-newsFree
007:	[archive.orst.edu]	alt.drugsFree
008:	[wais.oit.unc.edu]	alt.gopherFree
009:	[sun-wais.oit.unc.edu]	alt.sys.sunFree
010:	[wais.oit.unc.edu]	alt.waisFree
011:	[alfred.ccs.carleton.]	amiga-slipFree
012:	[munin.ub2.lu.se]	amiga_fish_contentsFree

```

013: [      150.203.76.2] ANU-Aboriginal-EconPolicies$0.00/minute
014: [   coombs.anu.edu.au] ANU-Aboriginal-Studies$0.00/minute
015: [   coombs.anu.edu.au] ANU-Asian-Computing$0.00/minute
016: [   coombs.anu.edu.au] ANU-Asian-Religions$0.00/minute
017: [      150.203.76.2] ANU-Australian-Economics$0.00/minute
018: [      150.203.76.2] ANU-CAUT-Academics$0.00/minute

```

<space> selects, arrows move, w for keywords, s for sources, ? for help

quake.think.com

Once you have logged onto the Quake machine, it will ask you for an optional "identifier" (e.g., your e-mail address) and then your terminal type. Accept the default terminal type of VT100. You will then be presented with an information screen. Go "q" to exit the screen and begin your search session.

The first thing you will see at this point is a menu consisting of a single meta-database called "directory-of-servers". Proceed as follows:

- ▶ Press <Spc> to select the meta-database (and marks it with an asterisk).
- ▶ Press <Return> to invoke the "Keywords:" prompt.
Type one or more keywords (use "^U" to clear the entry if you make a mistake).
- ▶ Press <Return> once more to initiate the search of the meta-database. If WAIS finds a match for your keywords in the titles or descriptions of the 500 or so databases at its disposal, it will display a menu of those databases.

- Now move the highlight bar over the found databases that interest you and press "u" for "use database." After a few seconds, WAIS will add the database to the universe of active databases in your current session. Repeat as desired to use other databases.
- Next press "s" for "sources" to return to the original screen of database sources in which you previously saw only the "directory-of-servers" meta-database. Now, however, you will see the other databases you have activated as well. Press "=" to clear the asterisk marking the meta-database (which you earlier selected). Then move the highlight bar to one or more of the new database(s) you have just added and press <Spc> to select it (mark it with an asterisk).
- <Return> at this point then initiates a keyword search of your selected databases (for details, see description of the Finland machine above).

You can quit WAIS as follows:

- "q" closes the connection with the remote WAIS client you have telneted and returns you to the *humanitas*> prompt.
- If you get stuck somewhere while using a remote WAIS client, use the standard telnet escape key, "^]", to reach the telnet> prompt. Go "quit" to get back to *humanitas*>.

Here is the beginning of a Quake WAIS session (with commentary in *italics*):

```
telnet quake.think.com
Trying 192.31.181.1 ...
Connected to quake.think.com.
Escape character is '^]'.

SunOS UNIX (quake.think.com)

login: wais
SunOS Release 4.1.3 (SUN4C-STANDARD) #9: Wed Oct 27 18:18:30 EDT 1993
You have new mail.
Welcome to swais.
Please type user identifier (optional, i.e user@host): [your e-mail address]
TERM = (vt100) <Return>
Starting swais (this may take a little while)...

This is the new experimental "wais" login on Quake.Think.COM

As the total number of sources has passed the 500 mark, we've found it's
become virtually impossible to find a source from the 25 screens of
sources.

I have decided that instead of presenting you with all the sources, I'll
just give you the Directory of Servers as a starting point. To find
additional sources, just select the directory-of-server.src source, and ask
it a question. If you know the name of the source you want, use it for the
keywords, and you should get that source as one of the results. If you
don't know what source you want, then just ask a question that has
something to do with what you're looking for, and see what you get.

Once you have a list of results, you should "u"se the result you desire.
You can "v"iew a result before you "u"se it, paying close attention to the
"description".

Please let us know how you like this approach by sending feedback to
"wais@quake.think.com".

- WAIS at Think.COM

/notice.text
q      [quits information screen]

#          Server          Source          Cost
001:  [ quake.think.com]  directory-of-servers  Free
```

<space> selects, w for keywords, arrows move, <return> searches, q quits, or ?*

<Spc> [select meta-database]

<Return> [initiate search of meta-database]

Enter keywords with spaces between them; <return> to search; ^C to cancel

Keywords: poetry <Return>

Searching directory-of-servers.src...

Initializing connection...

Found 6 items.

SWAIS			Search Results	Items: 6
#	Score	Source	Title	Lines
001:	[1000]	(directory-of-se)	POETRY-index28	
002:	[858]	(directory-of-se)	poetry18	
003:	[286]	(directory-of-se)	ANU-Buddha-L80	
004:	[286]	(directory-of-se)	ANU-Cheng-Tao-Ko-Verses99	
005:	[286]	(directory-of-se)	ANU-Dhammapada-Verses91	
006:	[286]	(directory-of-se)	ANU-Philippine-Studies92	

<space> selects, arrows move, w for keywords, s for sources, ? for help

u [use database at highlight bar]

Retrieving: POETRY-index

Gtng "POETRY-index.src" from directory-of-servers.src...

Adding Source: POETRY-index

s [go back to "sources" screen]

001: *	[quake.think.com]	directory-of-servers	Free
002:	[sunsite.unc.edu]	POETRY-index	Free

= [clear previous selection of meta-database]

[arrow down] [move highlight bar down]

<Spc> [select new database]

<Return> [initate search of database]

Enter keywords with spaces between them; <return> to search; ^C to cancel

Keywords:

CHAPTER 12:

ANONYMOUS TELNET

(AND HYTELNET SEARCH ENGINE)

12.1: THE TELNET CONCEPT

Telnet is the protocol that allows users to logon to a remote computer and use that computer's commands interactively. When you dial up Humanitas from home, for example, you are really telneting the Humanitas server by way of its old "hcfmail" name (the new terminal server installed on Sept. 20, 1994, accepts the commands, "connect hcfmail", "connect humanitas", and "telnet humanitas" as equivalents).

Because Telnet does not limit the user to being a passive "shopper" (to resume my previous metaphor), it is in principle a more powerful interface to the Net than FTP. But in practice, Telnet and FTP are roughly equivalent in power because users can usually telnet in fully-interactive mode only to local systems for which they have a userid and password (e.g., Humanitas). They then use the local system to telnet in turn to more restricted, remote Anonymous Telnet servers that require either no userid and password or only generic ones (like "guest," "wais," or "lynx" plus an e-mail address). Such Anonymous Telnet servers commonly run only one (or a few) specialized applications--each with its own, custom-designed menu of commands--into which the user is automatically dumped. The classic Anonymous Telnet application is a library catalog whose only commands are those allowing a user to select one of a limited number of databases (e.g., books or periodicals) and then

perform certain search and display operations. Anonymous Telnet is also used to make connections to public-access WAIS, Lynx, and other "remote local client" programs.

12.2: OPENING AND CLOSING A TELNET SESSION

To open a session with a remote Anonymous Telnet server, type `["telnet [site-address]"` at the `humanitas>` prompt.

Once the connection with the remote server has been made, there is subsequently no standard interface (it all depends on the particular library catalog or other resource you tap into). Therefore, I will proceed here purely by example. The following is the beginning of an Anonymous Telnet session with the Library of Congress online catalog (available M-F 6:30am-9:30pm, Sat 8am-5pm, Sun 1pm-5pm Eastern time):

```
humanitas> telnet locis.loc.gov
```

```
Trying 140.147.254.3 ...
```

```
Connected to locis.loc.gov.
```

```
Escape character is '^['.
```

```
  L O C I S :  LIBRARY OF CONGRESS INFORMATION SYSTEM
```

```
  To make a choice: type a number, then press ENTER
```

- | | | | |
|---|-----------------------------|---|-------------------|
| 1 | Library of Congress Catalog | 4 | Braille and Audio |
| 2 | Federal Legislation | 5 | Organizations |
| 3 | Copyright Information | 6 | Foreign Law |

```
* * * * *
```

```

7  Searching Hours and Basics
8  Documentation and Classes
9  Library of Congress General Information
10 Library of Congress Fast Facts
11 ** Announcements **      New Interface for Some Files]

12  Comments and Logoff
Choice:

```

At any time within a telnet session, you can use the escape character, "`^]`", to shell out to the `telnet>` prompt and access a few utility commands. (Your session with the remote machine will still be running in the background unless you shut it down as per below.) Useful options at the `telnet>` prompt include:

- "`close`" breaks any connection with a remote Telnet machine and leaves you at the `telnet>` prompt of the local-client telnet program on Humanitas. Use this option only if you get stuck in a telnet session somewhere (or use "`quit`"). If the session is running fine, it usually includes its own menu choice for logging off.
- "`open [site-address]`" instructs Telnet to open another connection. If you did not close your previous connection, "`open`" by itself (without a site-address) returns you to that session where you left off.
- "`quit`" breaks any connection with a remote Telnet machine and exits the local-client Telnet program as well, leaving you at the `humanitas>` prompt. (To open another connection at this point, you will need

"telnet [site-address]" rather than "open [site-address]".)

12.3: HYTELNET SEARCH ENGINE

Hytelnet is a database of telnet sites that allows you to search in hypertextual fashion for the names and site-addresses of library catalogues, online databases or bibliographies, electronic texts, and other Net resources accessible via telnet. To start Hytelnet, go "telnet access.usask.ca" at the humanitas> prompt. Login as "hytelnet".

The first screen that appears is a menu of choices (e.g., "Library Catalogs") with a highlight bar. As in the case of other line-mode hypertext interfaces (cf., Lynx below), navigating this and subsequent menus is a matter of moving the highlight bar and using a few one-letter commands:

- *<Up>, <Down> Arrow Keys move the highlight bar from menu item to menu item. Each item is a hypertext link whose activation will jump you either to another menu or to an information screen.*
- *<Right> Arrow Key follows a link forward. Thus if you place the highlight bar over "Library Catalogs" in Hytelnet's opening menu and go <Right>, you will jump to the first in a series of Library Catalog menus.*
- *<Left> Arrow Key follows a link backward. If you have jumped from the Hytelnet opening menu to the first*

Library Catalog screen, <Left> will retreat to the previous menu. Pressing <Left> repeatedly will allow you to backtrack through all the menus you have visited one after the other.

- ▶ "+" , "-" move to the next and previous page of the present menu, respectively.
- ▶ "m" returns to the Hytelnet Main (opening) menu from any point.
- ▶ "i" initiates a keyword search of all the menus and listings in the Hytelnet database. Use this option, for example, to find the names of all library catalogs that have the word "California" in the title. It doesn't matter where you are in Hytelnet when you use "i".
- ▶ "q" quits Hytelnet at any point.

By following the hypertext links, you will eventually come to a bottom-level menu of telnet resources (e.g., a menu of U.S. library catalogs). Following a link forward at this point brings up an information screen with a descriptive precis of the resource. The most vital piece of information in this precis is the telnet site-address of the resource. Hytelnet does not connect to site-addresses for you, so you will need to quit Hytelnet and use telnet from the *humanitas*> prompt.

The following is the opening of a Hytelnet session:

```
humanitas> telnet access.usask.ca
```

Trying 128.233.3.1 ...

Connected to access.usask.ca.

Escape character is '^['.

ULTRIX V4.4 (Rev. 69) (access.usask.ca)

login: *hytelnet*

Last login: Mon Sep 12 15:36:05 from adastratiac.net

ULTRIX V4.4 (Rev. 69) System #3: Fri May 27 13:38:04 CST 1994

Welcome to Ultrix V4.4 on access.usask.ca...

Welcome to HYTELNET version 6.7.x

Last update Sept. 11, 1994

What is HYTELNET?	<WHATIS>
Library catalogs	<SITES1>
Other resources	<SITES2>
Help files for catalogs	<OPOOO>
Catalog interfaces	<SYSOOO>
Internet Glossary	<GLOSSARY>
Telnet tips	<TELNET>
Telnet/TN3270 escape keys	<ESCAPE.KEY>
Key-stroke commands	<HELP>

.....

Up/Down arrows MOVE	Left/Right arrows SELECT	? for HELP anytime
m returns here	i searches the index	q quits

.....

HYTELNET 6.7 was written by Peter Scott

E-mail address: aa375@freenet.carleton.ca

Unix and VMS software by Earl Fogel

CHAPTER 13:

USENET NEWSGROUPS (nn NEWSREADER)

13.1: THE USENET CONCEPT

Usenet is a set of conventions for posting and circulating "news" (more accurately: "open letters," "discussions," "debates," or even "gossip") whose effect is to convert the Internet into a gigantic, world-spanning bulletin board cum soap box cum seminar. At last count, the Usenet is populated by nearly 7,000 so-called "newsgroups," some of which are read by thousands each day. Newsgroups are either moderated or unmoderated, and are organized into so-called "hierarchies": seven mainstream hierarchies, the notorious "alt" hierarchy, and assorted other hierarchies. "Hierarchy" refers to the fact that newsgroups are addressed according to a nomenclature resembling Linnaean classification: by the equivalent of kingdom, phylum, genus, and species. Thus the name, "rec.food.drink.beer", for example, means that the beer newsgroup is part of the recreation hierarchy, the food subhierarchy, and the drink sub-subhierarchy. (The "rec.food.drink.coffee" group is an evolutionary neighbor.)

The seven mainstream hierarchies are: **comp** (on computers and software), **news** (on the Usenet itself), **rec** (on travel, art, pets, music, tv, sports, books, etc.), **sci** (on science), **soc** (on societies and cultures), **talk** (on controversial issues), and **misc** (etc.). The "alt" hierarchy is definitely "alternative," featuring an eye-opening number of wild, glorious,

controversial, offensive, obscene, fetishistic, or transient interest groups. Other hierarchies include "bit" (where some Bitnet list server groups lead a second life), "de" (conducted in German), "relcom" (initiated in the USSR), and "k12" (for elementary and high school teachers/students).

Since more than a hundred new messages can be posted daily to a popular newsgroup (old messages are automatically deleted to make room), most users follow just a manageable number of groups by "subscribing" to them (a misleading term, since subscription requires only that users check off groups in a special file under their control). They then use a program called a "newsreader" (also called "news browser") to present them with newsgroup postings to be selectively read, junked, downloaded, or answered.

The machinery that makes all this happen has a decentralized nature that fits in well with the overall premise of the Internet. When a user posts a message to a newsgroup, the message is first of all stored on a UCSB computer that functions as the "newsfeed" (or news server) for Humanitas. (Presently, the newsfeed for Humanitas is the UCSB machine named "mercury.ucsb.edu", but this will shortly change.) Periodically, the newsfeed connects with newsfeeds elsewhere in the world to exchange, as it were, body fluids: the local machine gives the remote machine fresh postings; and, reciprocally, the other gives the local machine news from elsewhere. By capillary action, news thus spreads from machine to machine across the world (accelerated at points by "hubs" that propagate news en masse).

A message posted from UCSB will thus show up on most newsfeeds within a day or two.

To read the news, a user starts the newsreader on Humanitas, points to (or preselects) certain newsgroups, and peruses lists of messages. Humanitas physically stores only

these listings, which are periodically updated by its newsfeed; the newsfeed stores the actual messages. When the user then selects items from the listings and asks for them to be displayed or downloaded, the newsfeed sends along the requested file. Because storage is limited, the UCSB newsfeed (like most newsfeeds) carries only a subset of the total 7,000 newsgroups; and it automatically "expires" old messages after a week or so.

Mechanics alone, however, do not adequately characterize the Usenet. If the Internet has a "culture," then Usenet newsgroups are at the core of that culture. A whole code of conduct, tradition, lore, mythology, and even acronyms (e.g., IMHO = "in my humble opinion") has grown up to govern Usenet "netiquette"--to the point that a substantial fraction of messages in the more popular newsgroups is devoted to flaming advertisement-posting lawyers, "cross-posters" (i.e., people who waste global bandwidth by posting multiple copies of a message in many newsgroups), new college users crowding in each fall with clueless postings, and especially that untouchable caste, the much ridiculed "AOL'ers" or America Online members. So, too, the conduct code shows itself in the surprising number of messages posted in the mode of abjection: "Sorry for this question, but I'm a newbie. What is a FAQ? Apologies in advance. Please do not flame me."

Many aspects of this code, IMHO, are silly or cliquish and do not withstand close scrutiny (the world does not in fact break down into just a few stereotypes: the newbie, the AOL'er, the college student, etc.). But at its core are four Net commandments whose roots in social, political, and economic belief (particularly of the post-Enlightenment West) cannot be lightly contravened: Thou shalt not impede the right of others to freedom of speech or assembly (thus the recent "raid" of the alt.tasteless group upon the rec.pets.cats group was a no-no);

Thou shalt not be uninformed (don't ask a question without checking to see whether it has already been answered); Thou shalt not confuse "discussion" with "advertisement" (classifieds or personals are okay; unsolicited mass advertisements are not); and--the ultimate machinic imperative--Thou shalt not waste bandwidth (a silly posting becomes super-silly when occupying storage space on thousands of newsfeed machines around the world).

The following should flame-proof new Usenet users:

- ▶ Read first, post later. Follow a newsgroup long enough as a "lurker" or pure spectator to know the ropes before posting. Many users, indeed, do not post at all, or only to a few groups. If you have never posted before, try some test messages first in the alt.test group.
- ▶ Check the FAQ: Most newsgroups have a FAQ or "frequently asked questions" posting consisting of advice, policy, or general information about the group. Watch for the FAQ to make a periodic appearance in a newsgroup's listings. There is also a special newsgroup (news.answers) that collects FAQ's of other groups. (Also check out news.newusers.questions).
- ▶ Consider the E-Mail Option: When replying to a newsgroup posting, users have the choice of posting the response for all the world to see or sending a private, e-mail message to the author of the original posting. In many circumstances, the "closed"

rather than "open" response is more appropriate (and conserves bandwidth).

- If not guilt, then shame. If none of the above makes any impression on you, then you may be a "shame"-rather than "guilt"-culture person. If so, consider the following before shamelessly posting: you never know who's listening. There are ways to post messages through an "anonymous" cut-out, but otherwise professors may be overheard by their students, students by their professors or prospective employers, and everyone by their family.

13.2: SUBSCRIBING TO THE NEWS: THE ".NEWSRC" FILE

The first time you use a newsreader program on Humanitas (see below), it will automatically create a long system file in your home directory named ".newsrc" whose function is to register and track your participation in the newsgroups carried by your local newsfeed. Here, for example, is a small part of a .newsrc file:

```
Alt.culture.usenet:!
Alt.current-events.bosnia!
Alt.current-events.flood-of-93!
Alt.current-events.inet92!
Alt.current-events.somalia!
Alt.current-events.wtc-explosion!
Alt.cyb-sys!
Alt.cyberpunk:
Alt.cyberpunk.chatsubo!
Alt.cyberpunk.movement:
Alt.cyberpunk.tech:
Alt.cyberspace: 1-19262,19266-19268,19312,19316-19318,19324-19326
```

```

Alt.cybertoon!
Alt.dads-rights!
Alt.dcom.catv!
Alt.dcom.telecom!
Alt.dear.whitehouse!

```

A colon after a newsgroup name (":") means that the group is subscribed; an exclamation point ("!") means the group is unsubscribed. While you can (and will want to) read groups you are not subscribed to, the newsreader program "keeps track" only of subscribed groups. This means that it can automatically cycle through them one after the other and keep a log of messages you have previously read or displayed (which are elided from the listings the next time you read the news). In the above .newsrsrc excerpt, for instance, the newsreader has updated the subscribed "Alt.cyberspace" entry with the reference numbers of messages that have been read.

Because when it is first created the .newsrsrc file subscribes you to every possible group on the newsgroup, you will want to edit .newsrsrc to unsubscribe from all but the groups you actually want to follow on a regular basis. This means replacing thousands of colons with exclamation points. If you know your way around the vi text editor in Unix, you can do so fairly painlessly (not including, of course, the pain of learning vi). But if you don't know vi, then don't try to use the simpler pico text editor to change every line in .newsrsrc manually. Instead, download .newsrsrc (see §2.1-2.4 on downloading/uploading) and edit it in your word processor. Use the search-and-replace feature in your word processor to replace every ":" with a "!". Then selectively reintroduce colons for the groups you are interested in. When done, use your word processor to save the file as an ASCII file (most word processors have an option that allows you to do this) and upload it to Humanitas in place of

the old `.newsrsrc`. (You should first make a backup copy of the original `.newsrsrc` under a different name in your Unix home directory.)

The benefit of a `.newsrsrc` file that has been pruned is that you can now use your newsreader either to cycle through a sensible number of subscribed groups or (starting the newsreader with a special option) to "drop by" unsubscribed groups. The newsreader will not keep a history of the messages you read in unsubscribed groups, but that should not be a problem (since you would subscribe if you really wanted to track the group). Subsequent alterations to `.newsrsrc` can then usually be handled on newsgroup-by-newsgroup basis with commands available within your newsreader program itself. For example, you can subscribe or unsubscribe to individual groups while looking at their listings. (Note: when new newsgroups are created on the Internet and/or added to our local newsfeed, your newsreader will automatically subscribe you. Just use the commands in the newsreader to unsubscribe if you are not interested.)

13.3: STARTING THE "NN" NEWSREADER AND POINTING IT TO A NEWSGROUP

There are several well-known newsreader programs, including the older `rn` and the newer `nn`, `tin`, and `trn`. Currently, Humanitas has only `nn`, which is a powerful, flexible, and convenient newsreader with some of the "message threading" capability of `tin` and `trn` (meaning it organizes message and responses into threads of discussion). Only the main features and commands of `nn` can be covered here. (For



an extensive, on-line manual, start `nn` and type the command, `":man".`)

There are two usual ways to start `nn`, depending on whether you wish to go to your subscribed groups or to unsubscribed groups.

- To go to your subscribed groups, simply type `"nn"` at the `humanitas>` prompt. The newsreader will start and ask whether you wish to enter the group you last left off reading. You will see, for example: `"enter soc.culture.asian.american (117 unread?)"`. Answer `"y"` or `"<Spc>"` to enter that group.
- To go to unsubscribed groups (or to subscribed groups out of sequence), type `"nn -g -r"` at the `humanitas>` prompt. The newsreader will start and ask, `"Enter Group or Folder (+./~)"`. (Unless you are an advanced user, ignore `"Folder (+./~)"`.) Fill in the name of the newsgroup you want to enter and `<Return>`. The newsreader will then reply: `"Number of articles (juasne) (j)"`. Answer `"u"` to see a list of all unread articles or `"a"` for all articles (these will be the same if the group is unsubscribed).

Note: at the `"Enter Group or Folder (+./~)"` question you can fill in just the first parts of a hierarchy name followed by a period, then use the spacebar to cycle through all the possibilities in that hierarchy. Pressing `<Return>` at any point goes to the named group. For example, you can respond to the `"Enter Group or Folder"` message with `"soc.culture."`, then use the spacebar to cycle through `"soc.culture.af"`, `"soc. culture.afghanistan"`,

"soc.culture.african", "soc.culture.african.american",
etc.

 **Important:** There is another way to start `nn`, but users should avoid it if they value their privacy. Typing "`nn [newsgroup]`" at the `humanitas>` prompt will go to a newsgroup directly without `nn` asking interactively for further information. But any command entered at the `humanitas>` prompt can be spied upon by others via the notorious "`w`" command. This command displays a list of users currently active on the system together with the last command they entered at the `humanitas>` prompt. This means that if you start `nn` with the "`nn`" or "`nn -g -r`" options, your proclivities will not be exposed to prying eyes (the "`w`" command will show only "`nn`" or "`nn -g -r`"). But if you start up, for example, with "`nn alt.sex`", . . . (enough said).  Note that there are other times when you will want to be circumspect at the `humanitas>` prompt. If you've got a file named "`Prof_X_is_full_of_it`", there is a finite chance that Professor X may actually be watching when you type "`pico Prof_X_is_full_of_it`" to add your latest witticism. (Only what is typed at the `humanitas>` prompt will be exposed; nothing that happens once you have already entered into a process like Pico, Pine, FTP, `nn`, etc., will be visible.)

13.4 SELECTING MESSAGES (THE `nn` SELECT MESSAGES SCREEN)

Once you have pointed to a newsgroup and "entered" it, you will see the **nn** reader's Select Messages screen. This displays a menu of the current, unread messages in the newsgroup beginning with the oldest. For example:

Newsgroup: soc.culture.asian.american		Articles: 167 of 335/10
a Ira Smailer	52	>>>>Asian Women are most cher<>in relationships/w Whites
b Chi Ming Cheung	45	>Asian Dating Rituals (White guy doesn't understand)
c Chi Ming Cheung	20	>white males i met this summer...
d Phillip J. Perri	21	>>
e Julian L Chan	38	>>>
f Brian Robinson	45	>>>
g Brian Robinson	46	>>>
h Phillip J. Perri	52	>>>>
i Phillip J. Perri	51	>>>>
j EBOSE	14	In a little predicament about the SCAA-UCB get-together
k EBOSE	12	>hey all you "chock of my life" people!
l BLo	10	>
m EBOSE	8	>
n EBOSE	24	>Another offshoot of the "<>" thread! (Re: FOBs and Nerds
o wplatts	97	>>Call for votes etc.
p Julian L Chan	129	>>>
q dWiGht mIsHiMa	196	-
r wplats	40	>>>>
s Julian L Chan	30	>>>
- 17:06 -- SELECT -- help:? ----79%----		

The name of the newsgroup is shown at the top. Also at the top are numbers indicating (in this example) that there are 167 unread messages in the current group and 335 total unread messages in the user's 10 subscribed groups.

The body of the Select Messages screen displays information about particular messages. In menu item "b," for example, "Chi Ming Cheung" is the author's name or userid (some authors can create a pseudonym; others may use a special "anonymous" remailer); "45" is the number of lines in the message; and ">Asian Dating Rituals (White guy doesn't understand)" is the subject description. Items marked with one

or more ">" signs are part of a discussion thread. ">" means that this is not the original message on the subject but a response to it. (Thus Chi Ming Cheung is saying something to or about White Guy who doesn't understand Asian dating rituals.) ">>" means a response to a response, and so on. A hyphen or "-" means that the message has the same subject as the menu item above it but is not part of the same thread.

The line at the bottom of the screen shows the time (24-hour clock), how to get a help screen (type "?"), and where you are in the total, current listings for this newsgroup (in this case, the present screen is 79% of the way through the listings for soc.culture.asian.american). On the first and last screens of a listing, respectively, you will see "top" and "bottom." If the present screen shows all listings for a newsgroup, you will see "all."

Before reading or performing any other action on a listed message, you must first "select" it on the menu. You can navigate the menu and select messages as follows:

- <Spc>, <, Pressing the spacebar displays the next screen of listings if there
 \$, ^ is more than one screen. Pressing
 "<" displays the previous screen. To move to
 the last screen of listings (i.e., the most recent
 messages), use "\$". To move to the first
 screen of listings, use "^".
- [letter] To select a message, just press the menu
 letter associated with it. For example,
 pressing "b" in the example above selects Chi
 Ming Cheung on White Guy. A highlight bar
 (as shown here) appears over the selected

item. To select more than one message at a time, press another letter. (Some shortcuts: pressing the period selects the next message down; pressing "a-j", for example, selects the range of messages from "a" to "j." Pressing "=". selects all messages in the newsgroup; and "= [keyword]" selects all messages whose listing (author or subject description) contains the keyword. You can select as many messages simultaneously as you wish; messages selected on one screen of a listing stay selected when you move on to further screens.

If you change your mind after selecting a message(s), just press the letter of the message once more to deselect it. Press "~" to deselect all selected messages.

13.5 READING MESSAGES

Once you have 'selected' one or more messages in the listings of a newsgroup (and they are highlighted), you can read your selections by using any of the following options. Messages will be displayed starting with those earliest in the listings:

- ▶ "Z" Upper-case "Z" is the most common way of opening messages for reading. It displays your selected messages and subsequently returns you to the listings of the same newsgroup.

- ▶ "X" Upper-case "X" is the same as "Z," except that when you are finished reading your selections you are moved to the listings of the next subscribed group. (If you started nn with the "nn -g -r" option, you will be prompted for a newsgroup.)
- ▶ <Spc> If you are on the last screen of the Select Messages screen, pressing the spacebar will display any messages you have selected rather than advance you to the next screen of the listings.

Once you have asked nn to display your selected messages, there will be a short delay while the newsfeed retrieves the file from storage and transmits it to Humanitas and your screen. The following message informs you of the newsfeed's progress:

Connecting to NNTP server mercury.ucsb.edu ... ok (posting is allowed)

If an old message has been cancelled or expired (which often happens on the first screens of a message listing), nn will inform you and then go on to the next selected message or return you to the listings if there are no further selections.

If everything goes right, then the first message you have selected will be displayed as in the following example:

02 Sep 1994 19:58
 Erich Schneider: alt.cyberpunk Frequently Asked Questions list
 Archive-name: cyberpunk-faq
 Last-modified: 8/19/1994
 Frequently Asked Questions on alt.cyberpunk

Assembled by Erich Schneider (erich@bush.cs.tamu.edu)

Posted every two weeks

This is a FAQ list for alt.cyberpunk. It is inspired by, but is not a direct descendant of, the previous unofficial FAQ, originally compiled by Andy Hawks (who has left FAQ-building behind), and later edited by Tim Oerting (who has graduated from UW and can no longer be editor). I have been an alt.cyberpunk reader since 1988, and have seen many a FAQ get asked in my time. I am dedicated to answering your questions and keeping this document up to date and available. If you have comments, criticisms, additions, questions, or just general invective, send to erich@bush.cs.tamu.edu. Send to that address as well if you would like the latest version of this document, which is also available via anonymous ftp as

"bush.cs.tamu.edu:/pub/misc/erich/alt.cp.faq". The latest archived version is available as "rtfm.mit.edu:/pub/usenet/news.answers/cyberpunk-faq". There is also a version that has

-- 00:06 --alt.cyberpunk-- LAST --help:?--Top 4%--

Messages are displayed one screen at a time with a status line at the bottom indicating the name of the current newsgroup, which of your selected messages is being displayed ("LAST" means that this is the last or only selected message), and where you are in the message.

You can move through a displayed message using the following commands:

- **<Spc>** or The spacebar or down arrow moves to
the next screen

<Down Arrow> of the message.
- **<Backspace>**, Backspace or the up arrow moves
to the previous

or **<Up Arrow>** screen of the message.
- **\$** "\$" moves to the last screen of a
message.
- **^** "^" moves to the first screen of a
message.

- */[keyword]* *"/" followed by a keyword initiates a search.*
- *g[line number]* *"g" followed by a line number moves to that line.*

You can move between messages (or between messages and the Select Messages screen) using these commands:

- *n* *"n" at any point in a message displays the next selected message. If there is no other selection, "n" returns you to the listings of the current newsgroup.*
- *p* *"p" at any point in a message displays the previous selected message.*
- *=* *The equal sign while reading a message returns you to the listings in the Select Messages screen. If there were previous selections that were not read, they will still be highlighted.*

13.6 DOWNLOADING MESSAGES

You can download messages to Unix files in your storage space while either reading a message or viewing a newsgroup's

listings in the Select Messages screen. The same commands work in slightly different ways.

While Reading a Message:

To download a message while reading it, use these options:

- S Upper-case "S" saves the message including full header.
- O Upper-case "O" (the letter, not the zero) saves the message including only an abbreviated header.
- W Upper-case "W" saves the message with no header.

Once you have pressed one of the above keys, **nn** will respond by suggesting that you download the selected message to a subdirectory of your News/ directory (itself a subdirectory of your home directory) under a path and filename generated from the name of the current newsgroup. For example, if you are downloading a message in the alt.cyberpunk newsgroup, **nn** will respond: "Write on (+~|) +alt/cyberpunk". (The "+" stands for the path name of the News directory, which in my case is /home/ayliu/News. The full path and filename in this instance would thus be /home/ayliu/News/alt/cyberpunk.) You can accept the suggested name by pressing <Return> or you can change it and then <Return> (for convenience, I usually change all suggested paths so that files are downloaded to a single download directory; **nn** can also be configured to use a certain download name by default).. (Note: if after you download the message to a Unix file you intend to download it in a second

stage to your local or home computer, you should be mindful of the DOS "8.3" filename limitation. To avoid problems, change the Unix filename to no more than 8 characters and a 3-character extension.) Once you have told `nn` what filename to download the message to, it will prompt for a confirmation ("Create /home/ayliu/News/"[path/filename]"? Press "y" to confirm. The download process is then initiated. `nn` informs you of its progress as in the following example:

```
Write /home/ayliu/News/cyberFAQ Article (* +):
Processing 'alt.cyberpunk Frequently Asked Questions list'.../home/ayliu/News/cyberFAQ created:
537 lines written.
```

From the Select Messages Screen:

To download messages from the listings in the Select Messages screen, first select (highlight) the desired messages. Then use these options:

- **S** Upper-case "S" saves the selected messages including full header.
- **O** Upper-case "O" (the letter, not the zero) saves selected messages including only an abbreviated header.
- **W** Upper-case "W" saves selected messages with no header.

Once you have pressed one of the above keys, `nn` will respond by suggesting that you download the selected message(s) to a subdirectory of your News/ directory (itself a subdirectory of your home directory) under a path and filename(s) generated from the name of the current newsgroup.

For example, if you are downloading a single message in the alt.cyberpunk newsgroup, nn will respond: "Write on (+~|) +alt/cyberpunk". (The "+" stands for the path name of the News directory, which in my case is /home/ayliu/News. The full path and filename in this instance would thus be /home/ayliu/News/alt/cyberpunk.) You can accept the suggested name by pressing <Return> or you can change it and then <Return>. Once you have told nn what filename(s) to download the message(s) to, it will prompt for a confirmation ("Create /home/ayliu/News/"[filename]"? Press "y" to confirm.

You will next see the question: "* selected articles on this page, + all selected articles". If you press "*", nn will download only messages selected on the present menu screen (even if there are highlighted menu items on other screens). Pressing "+" downloads selected messages from all screens of the current listings.

The download process is then initiated. nn informs you of its progress as in the following example:

```
Write /home/ayliu/News/cyberFAQ Article (* +):
Processing 'alt.cyberpunk Frequently Asked Questions list'.../home/ayliu/News/cyberFAQ created:
537 lines written.
```

🐼 A particularly handy trick is to edit the "init" file in your .nn/ directory as follows. Change to the .nn directory ("cd .nn"); then call "init" into the Pico editor by going "pico init". Use Pico to add the following lines to the "init" file (keep the lines together):

```
save-files
alt* +x*
```

(Be sure to place an "end" at the end of the "init" file if it is not already there.) Now when you download a file from any alt group (represented by "alt*" above), nn will automatically save it under the name "x1" in your News/ directory. If you download another file, nn will use the name "x2", and so on. (Customize the "alt*" and "x*" in the above settings to designate the groups and filenames you want.) Now when you use Kermit to transfer the files in a second stage of downloading from Humanitas to your home or local pc, you can simply ask Kermit to "send x*". Kermit will send x1, x2, x3 (and so on) in automatic succession.

13.7 POSTING AND RESPONDING TO MESSAGES

When responding to or posting messages in a newsgroup, you have three basic choices: sending a private e-mail message to the author of a previous message, posting a public response to a previous message, or posting a message on a new subject.

Responding by E-Mail:

While in the Select Messages screen or while reading a message, you can respond privately to its author via e-mail by typing upper-case "R". If you use "R" from the Select Messages screen, nn will first ask, "Reply to author of article?" (answer with the menu letter of the message you wish to reply to). If

you are reading the message, **nn** will immediately ask "Include original article?" (this is also the next step when you start from the Select Messages screen). Answer "y" or "n" (if you decide to include the original message, it will be reproduced with lines beginning ">"; delete all material not germane to your purpose).

At this point, **nn** will cede control to your text editor to compose and edit your e-mail response (the header will already be filled in). When you are finished, exit your text editor normally (e.g., go "^X" in Pico). (The text editor will suggest a name in the /tmp directory to save your reply to before sending it out; accept the suggestion). **nn** will resume control and give you the following menu of choices:

a)bort e)dit h)old m)ail r)edit s)end v)iew w)rite
Action: (send letter)

Press <Return> or "s" for the default action of "send." Use "a" or "e" if you change your mind.

Posting a Response:

While in the Select Messages screen or reading a message, you can post a public response by typing upper-case "F". If you use "F" from the Select Messages screen, **nn** will first ask, "Follow Up to article?" (answer with the menu letter of the message you wish to reply to). If you are reading the message, **nn** will immediately ask "Include original article?" (this is also the next step when you start from the Select Messages screen). Answer "y" or "n" (if you decide to include the original message, it will be reproduced with lines beginning ">"; delete all material not germane to your purpose). At this point, **nn** will cede control to your text editor to compose and edit your response (the

header for the posting will already be filled in). When you are finished, exit your editor normally (e.g., go "^X" in Pico). (The text editor will suggest a name in the /tmp directory to save your reply to before sending it out; accept the suggestion). `nn` will resume control and give you the following menu of choices:

a)bort e)dit h)old m)ail r)edit s)end v)iew w)rite
Action: (send letter)

Press <Return> or "s" for the default action of "send." Use "a" or "e" if you change your mind.

Posting a Message on a New Subject:

At any point in `nn`, you can post a message on a new topic by typing upper-case `:post`. `nn` will ask, "POST to group". Fill in the name of the newsgroup you wish to post to.


`nn` will then lead you step by step through creating a header for your posting, prompting you for a subject, keyword(s), descriptive summary, and "distribution." Be succinct but clear in filling in the subject, keyword(s), and summary; these determine the message description that others will see when selecting messages to read. The "distribution" option allows you to specify whether you want your posting to be circulated worldwide or more locally (the most common local choices in our context: "na" for North America, "usa" for the U.S., "CA" for California, "can" for Canada). You can leave the distribution slot empty to accept the default "world" for worldwide.

Once the header is completed, `nn` will cede control to your text editor to compose and edit your posting. When you are finished, exit your text editor normally (e.g., go "^X" to exit Pico). The editor will suggest a file in the /tmp directory to save your message to before sending it out; accept the

suggestion. `nn` will resume control and give you the following menu of choices:

```
a)bort e)dit h)old m)ail r)edit s)end v)iew w)rite
Action: (send letter)
```

Press `<Return>` or `"s"` for the default action of `"send."` Use `"a"` or `"e"` if you change your mind.

 By default, `nn` is set to use the `vi` text editor to compose postings. (See the box at the end of §6.2 for some elementary instructions on `vi`.) Unless you are fluent in `vi`, however, you would be well advised to change the default to the Pico text editor. To do so, use Pico to open up the file named `"init"` in your `.nn/` directory and insert the line:

```
set editor pico
```

(Be sure to place an `"end"` at the end of the `"init"` file if it is not already there.)

13.8 NEWSGROUP HOUSEKEEPING (CLEARING, CATCHING UP, UNSUBSCRIBING, SUBSCRIBING)

Once you have finished reading, responding to, or posting messages in a newsgroup, you will want to move on to another

newsgroup or quit `nn`. Before doing so, however, you may want to do some housekeeping.

Clearing (Updating) Subscribed Newsgroups:

In the case of subscribed groups, `nn` by default marks as "read" any message that you display on the screen (i.e., it records the reference number for the message in your `.newsrsrc` file). Messages marked "read" are omitted from the listings the next time you start `nn`, though you can see all listings again by starting `nn` with the command, "`nn -x [newsgroup]`". It is often convenient, however, to ask `nn` to mark "read" all remaining, undisplayed items in a newsgroup because you are not interested in them and do not wish to plow through them again in the message listings the next time you check that newsgroup.

You can do so in either of the following ways:

- While in the Select Messages Screen, press upper-case "J". `nn` will respond:

Mark read S)een U)nmarked A)ll *(+selected a-z. [LN]
Use J repeatedly to select other functions

Press upper-case "A" to mark all listed messages as "read".

If there is more than one screen of listings in the current newsgroup, `nn` will ask "On all screens?" Answer "y".

- While reading a message (presumably the last message that interests you in the newsgroup), press upper-case "X". This will mark all articles as read and move you to the next of your subscribed newsgroups.

"Catching Up" in Subscribed Newsgroups:

Occasionally (e.g., after you have been away on a trip), you will want to clear your slate of the hundreds or even thousands of newsgroup postings that have accumulated in your absence. To do so conveniently, start `nn` with the `"-ao"` option (i.e., `"nn -ao"; "O"` here is the zero). `nn` will ask if you want to "catch up" on your newsgroups (i.e., mark them as "read") "automatically" or "interactively". If you choose `"i"` for interactive, `nn` will then ask you if you want to clear each of your subscribed groups one after the other. Choose `"y"` for yes to clear a group.

Unsubscribing/Subscribing to Newsgroups:

While reading a subscribed newsgroup (whether you are currently in the Select Messages screen or reading a message), you can unsubscribe by pressing upper-case `"U"`. `nn` will ask for a confirmation (answer `"y"`) before revising your `.newsrsrc` file so that you are no longer subscribed to the group. You will periodically need to use this command to unsubscribe from newly-created groups, which `nn` automatically subscribes you to.

If you the newsgroup you are perusing is unsubscribed, the `"U"` command will subscribe you.

13.9 CHANGING NEWSGROUPS AND QUITTING NN

After taking care of all business in a particular newsgroup, you can move to another newsgroup as follows:

If you started `nn` with "`nn`" in order to go to your subscribed groups:

Use one of these options while in the Select Messages screen:

- `<Spc>` While on the last screen of listings in a newsgroup, the spacebar takes you to the next of your subscribed groups.
- `"N"` Upper-case `"N"` moves to the next of your subscribed groups at any point.
- `"B"` Upper-case `"B"` moves back to the previous group in your sequence of subscribed groups.
- `"G"` Upper-case `"G"` goes to a group (subscribed or unsubscribed) out of sequence. `nn` will ask for the name of a group to go to.

If you started `nn` with "`nn -g -r`" in order to go to unsubscribed groups:

Use one of these options while in the Select Messages screen:

- `"N"` Upper-case `"N"` exits current group and causes `nn` to prompt for the name of the next group to go to.
- `"P"` Upper-case `"P"` goes to the previous group that you looked at in the current `nn` session.

Quitting `nn`:

When cycling through your subscribed groups, `nn` quits automatically after the last group. When viewing unsubscribed groups with the "`nn -g -r`" option, `nn` quits automatically if you don't specify any further group when prompted. (It also quits, unfortunately, if you make a mistake typing a group name).

To quit `nn` at any point (from any screen), press upper-case "`Q`".

CHAPTER 14:

GOPHER (AND VERONICA SEARCH ENGINE)

14.1: THE GOPHER CONCEPT

Originally created in 1991 at the University of Minnesota as a means of distributing online university information, Gopher has since become one of the great success stories of the Internet--expanding drastically beyond its initial uses and becoming (along with the more recent World Wide Web) one of the two all-purpose, easy-to-use Net interfaces. For most humanities users (who do not yet have the high-end hardware and software needed to access the World Wide Web fully; see below), Gopher will prove by far the single most useful Net tool.

Or rather, "tool" is too narrow a term. As indicated by its own metaphors (e.g., "tunneling" through "gopherspace"), Gopher is a whole habitat or environment. Or to use the metaphor of this manual, it and the Web are the only interfaces to the Net that come anywhere near to showing the whole landscape through the "windshield." Recently augmented with multimedia capabilities (but not on Humanitas, where "Gopher+" is not available), Gopher promises to engage the Web in a future contest for Net dominance.

Gopher is the Net's menuing environment par excellence. It takes the Net's various resources (not just files posted on Gopher servers but also FTP sites, Telnet sites, WAIS sites, various searchable indexes, etc.) and plugs them into a single,

yet surprisingly adaptable, format: the hierarchical menu. When you open the local-client Gopher program on Humanitas, it displays a top-level menu of the resources it can connect to. Choosing any item displays or "tunnels" to that resource, which in top-level menus will probably be whole, other Gopher sub-menus. Choosing an item on a sub-menu then tunnels to a sub-sub-menu . . . and so forth. At some point, you hit pay dirt in the form of content (e.g., a text file, a Telnet site, a searchable database). Gopher will then allow you to call up the content for viewing or other appropriate action. In the case of files, you can get a copy for yourself by asking Gopher either to e-mail it to you or to download it as a file to your Unix space on Humanitas.

To make life even easier, you can also customize your Gopher environment by creating your own favorite menu of resources (through the "bookmarks" feature).

14.2: STARTING (AND QUITTING) GOPHER

There are three ways to start the Gopher local-client program from the *humanitas*> prompt:

- "gopher" Typing "gopher" by itself starts Gopher with the default Gopher top-level menu for your server. In the case of the Humanitas server, the default is as follows:

```
--> 1.  About Letters and Science Gopher.
      2.  About The College of Letters and Science.
      3.  Computing and Networking/
```

4. Department and Class Information/ 5. Local Housing Information/ 6. Other Gopher Servers/
7. Students Academic Affairs/

Press ? for Help, q to Quit, u to go up a menu

Page: 1/1

This menu is good for UCSB-specific information but not very efficient for other purposes (though its submenus contains riches). You have to tunnel out through the "Other Gopher Servers/" item to get to the rest of the world.

- **"gopher -b"** Typing "gopher -b" starts Gopher with your custom "bookmarks" (a menu you have created by using Gopher's bookmark option to clip any item from any menu found in gopherspace). The bookmarks are stored in a system file in your home directory on Humanitas called ".gopherrc". If you don't have a .gopherrc (or only an impoverished one), root around for a good one you can borrow from someone else who has done a lot of

tunneling. The best one I have so far encountered for the humanities comes courtesy of Francis Steen (it is available on my *for_all_hands* public directory on Humanitas; see Appendix). Steen's current bookmarks span seven screens as follows:

- | | |
|-----|--|
| --> | 1. Virtual Reference Desk/
2. InterNIC Directory and Database Services (AT&T)/
3. Internet-wide e-mail address searches/
4. Veronica and Jughead Servers (search gopherspace)/
5. The Global Electronic Library (by Subject)/
6. CARL (Colorado Alliance of Research Libraries) <TEL>
7. Libraries/
8. Library of Congress Online Systems/
9. Online Manual Directory (Unix Commands)/
10. Blacksburg Electronic Village Gopher Server/
11. Washington University FTP Archives/
12. Video & Movie Review Database/
13. Dictionary Webster's Dictionary (searchable) <?>
14. Roget's Thesaurus <?>
15. Language and Literature Databases/
16. Databases at Michigan State /
17. Gopher Central at U Minn /
18. InfoSlug at UC Santa Cruz/
19. Internet file server (ftp) sites/
20. Directory of Scholarly Electronic Conferences (searchable) <?>
21. Search UseNet Movie Archives <?> <?>
22. USENET News (from Florida State)/
23. Electronic Books Directory (tons!)/
24. Usenet FAQs by Newsgroup Name/
25. Requests For Comments (Internet RFCs)/
26. English-server.hss.cmu.edu/
27. Cyber-stuff (cyberspace, cyberpunk)/
28. Gender and Sexuality Stuff (Eng. Server)/
29. Graphics Stuff (incl. .gifs, .gif archives, etc.)/
30. Queer Resources Directory/
31. Race Stuff (Eng. Directory)/
32. Discipline-specific research materials/
33. Science fiction reviews (search) <?> <?>
34. Clearinghouse of Subject-Oriented Internet Resource Guides/ |
| | 35. Film and Television Resources/Databases/
36. Internet Wiretap / |
| | 37. Information by Subject Area (good)/ |
| | 38. Carnegie Mellon University AC&M Gopher/
39. Modern Language Association Bibliography (trial test) <TEL> |

40. CUI's W3 Catalog
41. Internet User's Glossary (search) <?>
42. The Internet Resource Discovery Project (UMich)/
43. CNIDR (Clearinghouse for Networked Information Discovery and/
44. FreeNets/
45. Information Organization - by Subject (Gopher Jewels)/
46. University of Oslo, Norway - phonebook <CSO>
47. Local Times Around the World/
48. The Artwork (GIFs fr. AMANDA/U-Wisc. Madison)/
49. Billboard Charts.
50. MTV Gopher/
51. U. of Manchester UK (Actors/actresses/directors etc. <?>
52. East Asian Studies (Washington/Lee Univ.)/
53. Search a full text database of poetry (NCSU) <?>
54. French Language Gophers/
55. Latin American Network Information Center/
56. CICNET gopher server (Electronic Serials)***/
57. Arts and Leisure - The Pipeline/
58. Comics - Texas A&M/
59. Entertainment - AMI-A Friendly Public Interface Gopher/
60. Games- Connections to Game Servers - CNS, Inc./
61. Multimedia - Univ of Arizona, FRI Multimedia Server/
62. Pictures - AMI -- A Friendly Public Interface/
63. Pictures - ASCII - Stanford University Medical Center/
64. Pictures - CityNet, Wellington, New Zealand/
65. Pictures - Image and Multimedia Library - Victoria Freenet/
66. Pictures - OTIS: the Operative Term Is STIMULATE - Via The/
67. Sounds - San Diego State University/
68. Television & Film - Panix Public Access/
69. Virtual Reality - Via Sunsite Gopher Server/
70. Matrix Information and Directory Services, Inc. (MIDS),/
71. Gateways - FTP Services Worldwide - Cleveland State/
72. UCB Bisexual/Lesbian/Gay/Queer Gopher/
73. Good Movies Database (had Tsui Hark!) <?>
74. TV Series Episode Guides/
75. Movie Trivia/
76. Telephone Directories - N. American Colleges/
77. Bad Subjects (a student journal)/
78. Apple Computer Higher Education gopher server/
79. EDUCOM Documents and News/
80. Internet Society (includes IETF)/
81. SURAnet NIC/
82. Women's Wire Gopher/
83. Search FTP sites (Archie)/
84. Archie Gateways/
85. E-text archives at UMich/
86. Free Lance Philosophy/
87. WAIS, Inc. Gopher/

- 88. VERONICA: Search "the world" using Veronica/
- 89. Other Gopher and Information Servers/
- 90. University of California - Berkeley/
- 91. Gopher Central at WLU (great menus!)/
- 92. Software library/
- 93. University of California Gopher Servers/
- 94. Spr94 UCLA Schedule of Classes/
- 95. CAMPUS-WIDE INFORMATION SYSTEMS --CWIS's (non-gophers)/
- 96. Film/Video Materials for Programs on Asian-Americans (UMich).
- 97. Film/Video Materials for Programs on Hispanics (UMich).
- 98. World Wide Web (at CERN) <TEL>
- 99. Whole Earth 'Lectronic Magazine - The WELL's Gopherspace/
- 100. IBM ACIS Higher Education Information Server - IKE/
- 101. Academe This Week: The Chronicle of Higher Education/
- 102. Wired magazine/
- 103. Postmodern Culture/
- 104. Mother Jones Magazine/
- 105. Voice of America gopher /
- 106. UPI News/
- 107. USA Today/
- 108. Search Computer & Information Services Newsletter <?>
- 109. Current Time/Date.
- 110. Santa Barbara weather.
- 111. Root gopher server: humanitas.ucsb.edu/

Press ? for Help, q to Quit, u to go up a menu

Page: 7/7

- **"gopher [gopher-site-address]"** You can also gopher to a specific gopher server by adding the site address on the command line. For example, "gopher english.server.hss.cmu.edu" brings you to the superbly interesting English Server machine at Carnegie Mellon Univ. (Run by graduate students, this server also has a World Wide Web home page). The server you connect to in this way will display its own top-level menu. The first screen of the menu for the English Server, for example, is as follows:

- ```
--> 1. -New Stuff/
 2. About this server (24k).
 3. Art and Architecture/
```

4. Bad Subjects/
5. Books/
6. Calls for Papers/
7. Classes/
8. Conference Line <TEL>
9. Contributions/
10. Cultural Theory/
11. Cyber/
12. Drama/
13. Feminism/
14. Fiction/
15. Film and Television/
16. Gender and Sexuality/
17. Government, Law and Society/
18. Graduate Students Only/

Press ? for Help, q to Quit, u to go up a menu

Page: 1/3

To quit Gopher at any point, go "q". Gopher will ask for a confirmation before returning you to the *humanitas*> prompt.

### 14.3: NAVIGATING GOPHER MENUS

On a Gopher menu screen, the equivalent of a highlight bar or cursor is the arrow represented by "-->". Use the following commands to move the "-->" through the menu:

- ▶ <Up> and <Down>     The up and down arrow keys on your keyboard move the "-->" through the menu an item at a time.
- ▶ ">" and "<"     The ">" and "<" keys, respectively, advance or retreat a screen at a time if the menu has more than one screen. (You can also press "b" for the previous

screen or a menu or <Spc> for the next screen.)

- *"/[keyword]"*                      This command searches the menu items for the next occurrence of the keyword.
- *"[number]"*                      Typing the number of a menu item followed by <Return> moves the "-->" to that item. It then also automatically tunnels to that resource (and so anticipates the commands listed below).

Once you have positioned the "-->" next to a menu item you want to investigate, tunnel down as follows:

- *<Right>*                      The <Right> arrow key on your keyboard follows the menu item to where it leads. This is the best command to use, since it has the advantage of being consistent with the command used to follow a hypertext link in such Web interfaces as Lynx (see below).
- *<Return>*                      The <Return> key at this point has the same effect as the <Right> arrow key.

If you want to tunnel back up to the previous menu you were looking at, use these commands:

► **<Left>**

The <Left> arrow key on your keyboard regresses to the earlier menu item. This is the best command to use, since it has the advantage of being consistent with the command used to follow regress down a trail of hypertext links in such Web interfaces as Lynx (see below). Pressing the <Left> arrow key repeatedly slides you smoothly out of gopherspace the way you came in.

► **"u"**

Pressing "u" for "up" has the same effect as <Left>.

To jump back from any menu to your original, top-level menu, go "m".

## 14.4: READING AND DOWNLOADING FILES, ETC.

On a Gopher menu, each item has an indicator at its end signifying what kind of resource it is (most commonly: "/", "<TEL>", "<?>", "."):

- A slash means a submenu (e.g. "Bad Subjects/").
- "<TEL>" is a Telnet site.
- "<?>" is a searchable database.

- "." is a file (e.g., "Acker: Dead Doll Humility.")

As you tunnel from top-level menus down into gopherspace, you will reach submenus in which some or all of the items are not just further submenus but actual content. Positioning the "-->" by these items and then tunneling into them by going <Right> will have different results depending on the kind of resource it is. In the case of a Telnet site, for example, you will be dumped into a telnet session (e.g., an online library catalogue). A searchable database will commonly present you with a box on screen to enter a keyword. And a file will simply be displayed on screen in the *more* pager.

When you finish reading a file or interrupt its display with "^C", Gopher will present you with the following choice of actions at the bottom of the screen:

|                                                                        |
|------------------------------------------------------------------------|
| Press <RETURN> to continue, <m> to mail, <s> to save, or <p> to print: |
|------------------------------------------------------------------------|

- <Return> redisplay the Gopher menu on which you found the file.
- "m" allows you to mail the file back to yourself as an e-mail message (Gopher asks you for your e-mail address).
- "s" saves or downloads the file as a Unix file in your storage space on Humanitas (Gopher will suggest the name of the file).

## 14.5: BOOKMARKS

One of the handiest features of Gopher is the ability to create a customized "bookmarks" menu of the resources that especially interest you.

While the "-->" is next to an item of interest on any Gopher menu you have found anywhere in gopherspace, press "a" to add that item to your bookmark list. (If you don't yet have a bookmark list, Gopher will start one for you.) Pressing upper-case "A" adds not just an item but the whole menu it is found on (this option can also be used to add a customized search of a searchable database).

To use your bookmark menu, either go "v" (for "view" bookmarks) from anywhere within Gopher or start Gopher with the "gopher -b" command.

If you change your mind and want to delete an item from your bookmark list, display the bookmark menu, position the "-->" next to the offending item, and go "d" for "delete."


---

☹️ Unfortunately, the tools Gopher offers for managing a bookmark list are relatively primitive. You can add and delete items, but there is no way from inside Gopher to rearrange the order of items (which can be desirable when a bookmark list grows long). To do so, you will have to use a text editor like Pico to open up the .gopherrc file in your home directory where the bookmark list is kept. The beginning of such a file, for example, look like this:

---

---

Using the text editor, manually edit the file as you desire (following the format that you see).

You can also manually add a new item to a bookmark list simply by entering the required information (from the "Type=" line to the "#" delimiter). Guess at such information you may not have as the Port number (or leave it blank).  To find the Type, Name, Path, Host, and Port number of Gopher resources, you can peek behind the screen of any Gopher menu as follows. Position the "-->" in Gopher next to an item of interest, then press "=". This will show you the source information.

## 14.6 VERONICA SEARCH ENGINE

Just as FTP has its Archie, Gopher has its Veronica. Veronica is a search engine invoked from inside Gopher (as a menu item) that allows you to search for keywords in either the titles of individual resources ("title" here means what appears on the Gopher menu listing for that resource) or the titles of whole Gopher menus (or "directories"). Searching titles of individual resources, for example, would allow you to locate, say, Ian Balfour's essay on the Pep Boys (it's posted on the English Server and elsewhere); searching titles of Gopher directories, by contrast, might only locate Balfour or the Pep Boys if either ran

their own Gopher server and titled a menu, say, "Root gopher server: pepboys.ucsb.edu".

After tunneling into a Veronica-search menu item, you set up the keywords for your search and then connect to one of a number of special Veronica-servers. Much like Archie-servers, Veronica-servers are machines that periodically connect with all known Gopher machines to harvest lists of available resources. Your keyword search will sweep the lists stored on a Veronica-server for a match. When the search is complete, Gopher will display the results on screen in the form of a regular Gopher menu allowing you to read, mail, or download any found item as per normal.

To use Veronica, follow these steps:

- ▶ Navigate through the Gopher menus available to you (either from the default Humanitas top-level menu or from your bookmark list) until you find one or more menu items titled something like the following:

Search the world using Veronica <?>

Search gopherspace at NYSERNet using  
Veronica<?>

Veronica server at NYSERNet <?>

Search Gopher Directory Titles at PSINet <?>

The first variant here leads to a submenu of Veronica-servers. The latter variants connect to actual Veronica-servers (e.g., the NYSERNet or PSINet machines).

If you can't find a menu item for Veronica searching anywhere, then do the following: start Gopher from the humanitas> prompt with the command, "gopher peg.cwis.uci.edu". This will connect you to the Gopher

server at UCL. On the top-menu that appears, choose "11. The World/". On the sub-menu that next appears, choose "7. "Searching 'The World' with Veronica."

- ▶ Decide whether you wish to search the titles of individual resources (such searches are commonly described on menu lines as "search gopherspace at," "Veronica server at," etc.) or the titles of Gopher directories ("Search Gopher Directory Titles"). Then position the "-->" next to the server you wish to use and tunnel in with the <Right> key. (Note: during busy times of the day, you may have to try several servers to find one not overloaded. If you initiate a search and the server is too busy, Gopher will either tell you that access is denied or seem to freeze. In the case of a freeze-up, just "^C" out.)
- ▶ Gopher will then present you with a keyword-search box like the following:

A screenshot of a terminal window. At the top, a dashed line separates the header from the content. Below the header, the text "Words to search for" is displayed on the left side of a horizontal gray bar. To the right of this bar, there is a vertical cursor line. Below the gray bar, the text "[Cancel ^G] [Accept - Enter]" is centered. At the bottom, another dashed line separates the content from the footer.

Fill in one or more keywords in the highlighted area. Veronica is case-insensitive; wildcards ("\*") and Boolean rules can be used. By default, Veronica is set to find up to 200 matches. If you want fewer or more matches, fill

in the highlighted area with your keywords followed by "-m[number]". "-m800", for example, searches for up to 800 matches. When you are done entering your keyword(s), press <Return> to initiate the search.

- If the search succeeds, Gopher will present you with a menu of the results. For example, a search of individual resources for the keyword "balfour" turns up the following menu:

```
--> 1. BALFOUR, JAMES.
 2. BALFOUR, JAMES.
 3. The_living_soil_E.B._Balfour._Ill_fares_the_land_Walter_Yellowlees...
 4. The_living_soil_E.B._Balfour._Ill_fares_the_land_Walter_Yellowlees...
 5. The_living_soil_E.B._Balfour._Ill_fares_the_land_Walter_Yellowlees...
 6. The_living_soil_E.B._Balfour._Ill_fares_the_land_Walter_Yellowlees...
 7. The_living_soil_E.B._Balfour._Ill_fares_the_land_Walter_Yellowlees...
 8. Balfour,-Evelyn-Barbara,1898-Yellowlees,-Walter./
 9. Stafford Jr, Earl B / Program Coord Balfour Academy.
 10. Balfour, Edward (1886) "Indian Forestry." Indian Forester 12:250...
 11. Alan Balfour's address at the AA London!
 12. Lesley Balfour - Wedding Presentation.
 13. Lesley Balfour - Wedding Presentation.
 14. Ian Balfour: Revolutions per Minute or the Pet Shop Boys Forever.
 15. Ian Balfour: Revolutions per Minute or the Pet Shop Boys Forever.
 16. Ian Balfour: Revolutions per Minute or the Pet Shop Boys Fore.. <HQX>
 17. A-Balfour.txt.
 18. balfour.uue.
```

Selected item 14 from this menu would then allow you to read or download Ian Balfour's Pet Shop Boys article.

## CHAPTER 15:

# WORLD WIDE WEB (Lynx and Mosaic Browsers)

## [Incomplete]

*[Author's Note: This is an incomplete, preliminary sketch for a chapter on the Web.]*

The World Wide Web is the hypertext and hypermedia segment of the Internet on which an increasing number of scholarly and commercial resources--complete with pictures (and even video and sound)--are now being posted. It is the single, fastest growing part of the Internet. However, access to the Web (or WWW) in its full, multimedia splendor requires more hardware and software than most Humanitas users presently command. You need a graphical-user-interface "front end" program on your local computer called Mosaic for Windows (or Mac). This program can be downloaded for free via FTP from the NCSA (at <ftp.ncsa.uiuc.edu>; in the directory /PC/Mosaic). But to run it you need a direct hookup to the campus broadband, at least a 386 processor (or its Mac equivalent), 4Mb of RAM, Windows 3.1, a stack program (e.g., Trumpet Winsock), and the Win32s 32-bit extender.

To access the Web, therefore, most Humanitas users will need to fall back to a text-only Web browser (sorry, no images or formatted text). The best such browser is the hypertextual Lynx program, which displays text with hypertext links in a relatively intuitive interface. Invoking a link jumps you from your current document to another specified document (which

can be on the same server computer or another server halfway around the globe).

Currently, Humanitas does not have a local Lynx client (though there are plans; type "lynx" at the humanitas> prompt to check if we have the program yet). However, you can use Lynx remotely through public access sites at Cornell and Univ. of Kansas. Either use my handy GOWEB program (see Appendix) or do the following:

1. At the humanitas> prompt, go "telnet  
fatty.law.cornell.edu" (or "telnet  
ukanaix.cc.ukans.edu")
2. When the telnet host lets you in, it will ask for a login.  
Login as "www"

Once you have logged in, the so-called "home page" of the WWW server will appear. From this point on, the objective is to navigate to whatever you want to see on the Web or (because the Web can also access other sorts of resources) on the Internet generally. The coordinates that the Web uses for navigation are "URL's" or "Uniform Resource Locators"--an addressing scheme that is complex but has the virtue of being standard for all types of resources. For example, a document created specifically for the Web has the URL address `http://host_name/directory_name/ document_name.html`. Or again, an ftp site accessible through the Web would have the address `ftp://ftp_site/directory_name/file_name`.

If you don't know where you want to go and just want to explore, then simply follow the hyperlinks offered on the home page of the Web server you have logged onto. Since the Web is

indeed a world wide one, it should be theoretically possible to start anywhere and end up by blind luck anywhere else.

However, if you are looking for a (a) specific site or (b) kind of resource, then do this:

(a) On the Cornell server, the "g" command will allow you to go to any specific URL address you know. For example, specifying "http://jefferson.village.virginia.edu/rossetti/fullarch" jumps you to the Dante Gabriel Rossetti Hypermedia Archive (by Jerome McGann).

(b) On the Kansas server the "g" command is disabled. However, there is a subject search-engine. Use the up and down arrow keys to move the highlight block to the "by subject" hyperlink on the Kansas home page. Press <Return> or the right arrow key. This will take you to a screen with the highlight pre-positioned over "Index." Hit <Return>. Now you will be in a search screen whose central feature is "SUBMIT \_\_\_\_\_" (Rather imperious!). Put the cursor in the blank and fill in what you want to search for. For example: rossetti. Do not hit <Return> at this point. Instead, move the cursor (or highlight block) to the "SUBMIT" and <Return>. This will perform the search and display the results at the bottom of the screen in the form of a hyperlink(s). A search for "rossetti", for example, gives you "Institute for Advanced Research in the Humanities" (or something like that), which is the server on which McGann's state-of-the-art Rossetti Archive resides at UVA. Finally, hit the hyperlink you have found and it will take you to the appropriate place on the Web.

Once you have zoned in on a resource you wish to explore, follow the hyperlinks it provides. For example, when you reach the UVA "Institute for Advanced Research in the Humanities" home page, you can hit "Publications of the Institute." This will bring up another set of choices. From these choices, hit the "Rossetti Archive." This should bring up the McGann work. From there, follow the hyperlinks where your heart lists.

The one great disadvantage of using a remote Lynx (besides the fact that remote sites may not let you in when they are overloaded) is that you cannot create a "bookmark" list of favorite Web sites for yourself. Nor can you change default configuration settings for your personal Lynx interface. Both these options require you to set up special Lynx system files; and you do not have file-writing permission on someone else's computer.

## APPENDIX I:

### MODEMS AND COMM PROGRAMS: A PRIMER

- ▶ Purchase a 2400 or 14,400 baud modem (currently about \$50 and \$100 by mail order, respectively). If you have no experience with modems and are trying to decide how much to invest, take my advice: go for the highest-speed modem currently in mainstream usage even if you have to save up for it (presently the front line is 14,400; within two years it will be 28,800). The difference between a 2400 and 14,400 modem is magical.
  
- ▶ You'll also need a communications program to run your modem (many modems come bundled with a basic comm program). Popular IBM-universe choices include such shareware products as MS-Kermit for DOS or Kermit for OS/2 (Kermit has only primitive interface amenities but is the fastest-gun-in-the-west when it comes to downloading/uploading files from our UCSB systems [see below]) and such commercial products as Procomm, Crosstalk, Smartcom, Bitcom, HyperAccess, etc. (most available in both DOS and Windows versions; Hyperaccess is also available for OS/2). The best commercial products have such user-friendly features as scroll-back (to review text that has already gone off the screen), automated learning of logon scripts, multiple file-transfer protocols, capture files (to save your whole session to a text-file that can later be edited and printed), display of graphics files, etc. Be sure that the program you purchase includes the Kermit file-transfer protocol, since Kermit is necessary to download/upload from our campus servers.

Communications programs usually have a "main" screen or menu from which you configure your communications port, set your

modem ATDT codes and other parameters, create a list of frequently-dialed numbers, etc. They also have a "communications session" screen that displays the actual communications once you have connected with the computer on the other side of your phone line.

To set up your comm program to connect your modem to the campus modem banks (which in turn feed into the campus server computers), you need to take the following, one-time-only steps:

--Configure your communications port by finding your way from the main screen of your comm program to the program's settings for "port," "interrupt," and "address." Make sure these correspond to the hardware and operating-system settings you chose--or accepted by default--when you originally installed your modem (usually by physically setting "jumper" or "DIP" switches on the modem). Normally, the first four hardware ports on an IBM-compatible computer have default interrupts and addresses as follows:

| <u>Port</u> | <u>Interrupt (IRQ)</u> | <u>Address</u> |
|-------------|------------------------|----------------|
| COM1        | 4                      | 3F8            |
| COM2        | 3                      | 2F8            |
| COM3        | 4                      | 3F8            |
| COM4        | 3                      | 2F8            |

Note that COM1 and COM3 share interrupts and addresses, as do COM2 and COM4. This can lead to many uncanny problems if you assign a modem to a port whose interrupt and address are already used by equipment on a "doppelganger" port (a mouse, for example, is usually assigned COM1, making the use of COM3 problematic). Set your modem to a port, interrupt, and address that are free. Usually, your comm program will come with a utility program called Zport or

something similar that scans all ports and recommends free assignments. More advanced, recent computers give you extra options (ports above COM4 as well as other possible interrupts and addresses).

- Once your port has been configured, then find your way from the main menus of your comm program to the settings that establish the initialization and dialing instructions for your modem. Modems must be "initialized" before each session via an "AT" or Hayes-compatible code that looks something like the following (depending on the particular modem):

ATQ0&C1&D2&K3 (consult your modem or comm program manual). Once the modem is initialized, then a dialing instruction can be sent (usually automatically from your communication program) by means of the code ATDT followed by the phone number. "ATDT" means "attention dial tone."

- The above set-up procedures are "global" in the sense that your communications port and modem initialization/dialing instructions are standard for all your communications sessions.

In addition, you must set certain other communication protocols governing your session with any particular server. To connect properly with the UCSB modem banks and servers, use the main menu of your comm program (and its appropriate submenus) to define a connection named, for example, "Humanitas". Your comm program will allow you to set various protocols and options for the Humanitas connection. The crucial settings are as follows: 2400 or 14000 baud (depending on your modem), 7-bit (as opposed to 8-bit), parity=none, stop bits=1, and emulation=VT100. If your program asks about "echo," set it to none (alternatively, it may ask about "full or half duplex"; set duplex to full). Some

programs also allow you to set a default file-transfer or downloading/uploading protocol; set this protocol to Kermit. This is also the occasion to set the proper phone numbers for the campus modems: 893-8400 for the 2400-baud modem bank or 893-3000 for the 14400-baud modem bank.

Once both your port and modem are set up, then in the future you can initiate a communications session simply by instructing your comm program to dial Humanitas (or MCL).

## APPENDIX 2:

### ALAN LIU'S *for\_all\_hands* DIRECTORY

I have created a directory in my home space accessible to all Humanitas users (users on other servers can ask me to e-mail materials to them). The directory is `/home/ayliu/for_all_hands`. Below is the directory's current "readme.txt":

(readme.txt updated 8/20/94)

Welcome to *for\_all\_hands*, Alan Liu's public access space! To see what is currently in this space, type `"ls -a /home/ayliu/for_all_hands"` at the *humanitas*> prompt. Copy any file you wish to your home directory by typing `"cp /home/ayliu/for_all_hands/[filename] [filename]"`. To display the contents of a text file (`__.txt` or `__.info`), type `"pg [path/filename]"` or `"more [path/filename]"`. (Ignore asterisks (\*) in any filenames you see; they just mean the file is executable.) Some of the menus included in my shell script programs cross-refer to other programs or files in *for\_all\_hands*. So if anything seems to be missing, just come back and pick it up, too.

Currently, *for\_all\_hands* contains the following programs or text-files:

(1) *do* (v. 1.1) (last updated 8/14/94)

This is a shell script program that serves as a beginner's interface to the Internet. Typing `"do"` (followed by <Return>) at the *humanitas*> prompt calls up a menu of options (including such Internet features as *gopher*, *ftp*, *archie*, *wais*, etc. Typing `"do [menu number]"` initiates the chosen action. See the `"do"` menu for more instructions.

(2) *goftp* (v. 1.0) (last updated 8/3/94)

This is a shell script program that automatically ftp's to selected ftp sites from a menu (the program comes with preselected sites useful for locating and downloading IBM-PC software, including Windows sites, OS/2 sites, Mosaic sites; you can customize the program with your own sites by editing it in *pico* or *vi*). Typing "*goftp*" at the *humanitas*> prompt brings up a menu of options. Typing "*goftp* [menu number]" initiates the chosen ftp action. See the "*goftp*" menu for more instructions.

(3) *.gopherrc* (last updated 8/14/94)

This is the gopher "bookmark" file I am currently using. It's the best I have so far run into: essentially a bottomless resource of gopher menus and submenus with a strong humanities accent. (Note: before copying

*.gopherrc* to your home directory, use the "*ls -a*" command to be sure that you don't already have a *.gopherrc* that you want to keep. If you already have a gopher bookmark file, then use the "*mv [original\_name] [new\_name]*" command to rename it before copying mine; otherwise mine will overwrite yours.) To use *.gopherrc*, type "*gopher -b*" at the *humanitas* prompt. Or, if you are already in gopher, type "*v*" (for "view bookmarks"). You can add your own items to *.gopherrc* using the "*a*" command while in gopher.

(4) *world\_wide\_web.info* (last updated 8/20/94)

This is an information file with instructions for accessing the World Wide Web via the Lynx text-only browser. Lynx doesn't give you the

multimedia view of the Web offered by Mosaic or another graphical interface, but it can take advantage of the Web's hypertext capability. It is a good way to access the Web for users without a direct hardwire

link to the campus broadband and heavy-duty hardware/software (meaning, almost all Humanitas users).

(5) goweb (v. 1.1) (last updated 8/20/94)

This is a shell script program that calls up a menu of text-browsers for the World Wide Web (two remote Lynx browsers and two flavors of the Cern browser). Typing 'goweb' at the humanitas> prompt brings up the menu with instructions; typing 'goweb [menu number]' telnets to a remote host offering public access to a Web browser. Login as "www".

## **INDEX OF KEY TERMS (DOUBLE-UNDERLINED TERMS) USED IN A DEFINING CONTEXT**

This is a "cheat" index mediating between an index proper and a glossary. References are to the first use of key terms in a defining context. (Note: this feature is not fully implemented in version 1.0 of this manual.)

|                                  |    |
|----------------------------------|----|
| Accepted Use Policies (AUP)..... | 48 |
| Anonymous FTP .....              | 73 |
| Archie search engine.....        | 81 |
| Archie servers.....              | 81 |
| Archived ("tar") files .....     | 39 |
| ASCII .....                      | 24 |
| Blocks.....                      | 23 |
| C-Shell.....                     | 23 |
| Client/Server .....              | 50 |
| Compressed Files .....           | 39 |
| Directories.....                 | 24 |
| Home directory.....              | 25 |
| Working directory.....           | 26 |
| Downloading .....                | 9  |
| FAQ.....                         | 94 |
| Files.....                       | 24 |
| Binary files.....                | 24 |
| Compressed files .....           | 39 |
| System files.....                | 27 |
| "Tar" (archived) files .....     | 39 |
| Text files.....                  | 24 |
| Uuencoded files .....            | 38 |
| FTP.....                         | 73 |

|                                             |        |
|---------------------------------------------|--------|
| Anonymous FTP .....                         | 73     |
| FTP sites, servers.....                     | 50, 73 |
| Gopher sites.....                           | 50     |
| Hierarchies (Usenet).....                   | 93     |
| Internet.....                               | 47     |
| Kilobyte .....                              | 23     |
| Mirror sites.....                           | 73     |
| "Netiquette" .....                          | 48     |
| Newsfeed (Usenet).....                      | 93     |
| Newsgroups (Usenet).....                    | 50, 93 |
| Newsreaders (Usenet).....                   | 93     |
| News "threads" (Usenet).....                | 96     |
| Packet-Switching.....                       | 47     |
| Password .....                              | 1      |
| Path.....                                   | 25     |
| Absolute.....                               | 25     |
| Relative.....                               | 25     |
| Permissions.....                            | 41     |
| "Regular expressions" .....                 | 82     |
| Root directory.....                         | 25     |
| Server/Client .....                         | 50     |
| Server computer .....                       | 1      |
| System Administrator (Sysadmin, Sysop)..... | 50     |
| "Tar" (archived) files .....                | 39     |
| TCP/IP protocol.....                        | 47     |
| Telnet.....                                 | 89     |
| Anonymous Telnet.....                       | 89     |
| "Threads" (Usenet).....                     | 96     |
| Uploading .....                             | 9      |
| Unix operating system.....                  | 23     |
| Usenet.....                                 | 93     |
| Hierarchies.....                            | 93     |

|                                |        |
|--------------------------------|--------|
| Newsfeed.....                  | 93     |
| Newsgroups.....                | 50, 93 |
| Newsreaders.....               | 93     |
| "Threads" of discussion.....   | 96     |
| Username, Userid.....          | 1      |
| Uuencoding.....                | 38     |
| VT100 Emulation.....           | 3      |
| WAIS search engine.....        | 81     |
| World Wide Web.....            | 50     |
| "Zip" (compressed) files ..... | 39     |