I
Import Bookmarks 20
inetd 56
info.anu.edu.au 52
ISO Latin-1 22, 25
ISO-Latin-1 57
Item Attributes 20

J
JPEG 17

K
Kanji 22
Kuypers, Jean-Pierre 25

L
libbtree.a 37
libftw.a 38
libtext.a 37

M
MacWrite II 22
Maes, Pascal 25
Makefile.config 54
Monaco 18
MPEG 3
MS-DOS 17

N
NCSA Telnet 17
NeXT 36
NeXTtext.tar.Z 37

P
pandaa.uiowa.edu 52
ph 53
PICT 17
Pirard, Andre 25
PKTMUX 56
Pomes, Paul 53
Preferences folder 23

Q
Quicktime 3

R
Recent menu 19
rtfm.mit.edu 51

S
SitExpan 16
SLIP 21
StuffIt 16
svc.conf 55
System 14
System 7 14
System Folder 23

T
TN3270 17
tolten.puc.cl 52
TurboGopher 13
TurboGopher Settings 23

U
UPI news 53

V
veronica 57

W
WAIS 55
wais-8-b5.1.tar.Z 38
waisindex 36
waissearch 38
waisserver 38
Webster 36
WorldScript 22

X
XCMD 26

Index

Symbols
/etc/svc.conf 55
“q 53

A
AdmitOne 19
Alternate Views 17
anonymous ftp 37
Another Gopher 22

B
Bookmarks 20
boombox.micro.umn.edu 12

C
Cancel All Requests 21
Compactor 16
consultant.micro.umn.edu 52
CptExpand 16
CSO 53

D
Dangling file 56
Digital Bookshelf 36
DownLine 19

E
ecnet 52
ecnet.ec 52

F
Finder 17
FTP 51
ftp.bio.indiana.edu 38
ftp.think.com 38

G
Get Attribute 19
GIF 17
Gilbert, Don 38, 40
gindexd 53
gopher.chalmers.se 52
gopher.sunet.se 52
gopherdist 57
gopher-faq 51

H
helper application 17

Path=1/veronica
Host=futique.scs.unr.edu

Is there somewhere I can retrieve a list of announced gopher links? I'd like to keep a local, up-to-date list of available gopher holes without requiring our users to gopher to umn just to scan GopherSpace.

In the Unix client/server distribution is a perl script called "gopherdist". Gopherdist can fetch the contents of any point in GopherSpace.

To dump the contents of all the North American links from gopher.tc.umn.edu do the following:

% gopherdist gopher.tc.umn.edu 70 "1/Other Gopher and Information Servers/North America" > .Links

Why doesn't my unix gopher client display ISO-Latin-1 characters properly?

It is the client's problem, the server is perfectly 8-bit transparent. The BSD curses library uses bit 8 in order to remember, whether a character has been displayed reverse. So use just /usr/5bin/cc and you get the System V curses version which is 8 bit clean.

Note that this may be a problem under other versions of UNIX too...

What is veronica?

veronica: Very Easy Rodent-Oriented Net-wide Index to Computerized Archives.

veronica offers a keyword search of most gopher-server menu titles in the entire gopher web. As archie is to ftp archives, veronica is to gopherspace. A veronica search produces a menu of gopher items, each of which is a direct pointer to a gopher data source. Because veronica is accessed through a gopher client, it is easy to use, and gives access to all types of data supported by the gopher protocol.

To try veronica, select it from the "Other Gophers" menu on Minnesota's gopher server, or point your gopher at:

Name=veronica (search menu items in most of GopherSpace)
Type=1
Port=70

When linking the Unix gopher server with WAIS I get undefined symbols, such as:

log_file_name logfile PrintStatus find_value Sources NumSources

This happens if you make gopherd before linking in the WAIS ir/ui directories. The fix is to "make clean" or remove gopherd/{waisgopher.o,Waisindex.o}and then remake gopherd. Or link the ir/ui directories first.

Why don't my WAIS indexes work? I never get anything back for searches. or Why do I get "Dangling file" error messages in my logfile?

The problem could be in the server. The server should be run using the -c option if you want WAIS to work. Another solution is to patch the WAIS code so that it doesn't check the files on the disk. Search the gopher-news archive for "dangling". This will turn up a single document with the patch.

My gopher server doesn't work under inetd, why?

It could be that your inetd server only supports a limited amount of arguments. For instance, the maximum number of arguments to an inetd server is 5. You can get around this by combining arguments: i.e.

```
gopherd -I -c
```

becomes:

```
gopherd -Ic
```

You may also leave the port specifier off of the command line. The gopher server automagically finds out the port it's running on.

Help! I have PC-NFS and want to use the PC-Gopher client. How?

Use a piece of software called PKTMUX, available at fine ftp sites everywhere. This will let you use any packet driver application.

Or, aquire a client that supports PC-NFS.

(You can subscribe to the Internet Society News by sending e-mail to isoc@nri.reston.va.us)

The Internet Gopher Protocol, Proceedings of the Twenty-Third IETF, CNRI, Section 5.3

Internet Gopher, Proceedings of Canadian Networking '92

The Internet Gopher, INTERNET: Getting Started, SRI International, Section 10.5.5

Tools help Internet users discover on-line treasures, Computerworld, July 20, 1992

TCP/IP Network Administration, O'Reilly.

Balakrishan, B. (Oct 1992) "*SPIGopher: Making SPIRES databases accessible through the Gopher protocol*". SPIRES Fall '92 Workshop, Chapel Hill, North Carolina.

Tomer, C. *Information Technology Standards for Libraries*, _Journal of the American Society for Information Science_, 43(8):566-570, Sept 1992.

On a DECstation I get the error message "/etc/svc.conf no such file or directory" when running the gopherd server, why?

A14: This is caused by the chroot() call in gopherd. It can be easily fixed by running gopherd with the -c option.

Alternatively you can copy /etc/svc.conf into a directory named "etc" inside the gopher-data directory.

The boolean searching terms don't work for my full-text index, why?

This is probably because the searching is being provided by WAIS. WAIS opts to return all documents that contain a search phrase within certain limits. WAIS searches do return the documents with the highest "score" at the top, those documents will have the closest relevance.

Alternatively you could get a booleanized version of wais from ftp.bio.indiana.edu.

When I try to build the UNIX software I get an error from make: "Must be a separator on rules line #. Stop" Why?

This is a problem with older makes that don't understand the "include" keyword. One easy way to cope with this problem is compiling GNU make, which does understand the include keyword.

If this is too difficult, remove the line:

```
include Makefile.config
```

from all the Makefiles and paste in a copy of Makefile.config at the top of each Makefile.

Or, instead of pasting you can make the client/server by going into the appropriate directory and typing:

make -f ../Makefile.config -f Makefile

Are papers or articles describing Gopher available?

Gopher has a whole chapter devoted to it in :

The Whole_Internet Users Guide and Catalog by Ed Krol (publisher O'Reilley & Associates, Inc; ISBN: 1-56592-025-2). (Editors note: ...Great book, go out and buy a bunch!)

The Internet Passport: NorthWestNet's Guide to Our World Online" By Jonathan Kochmer and NorthWestNet. Published by NorthWestNet, Bellevue, WA. 1993. 516 pp. ISBN 0-9635281-0-6. Contact info: passport@nwnet.net, or (206) 562-3000

A Students Guide to UNIX by Harley Hahn. (publisher McGraw Hill, Inc.; 1993 ISBN 0-07-025511-3)

Other references include:

The Internet Gopher, "ConneXions", July 1992, Interop.

Exploring Internet GopherSpace "The Internet Society News", v1n2 1992,

How can I set up a "CSO" phone book server? Where is the software?

CSO phone book servers are also known as "qi" servers. The software implementation can be gotten via anonymous ftp from

```
uxc.cso.uiuc.edu (128.174.5.50)
```

as

```
/pub/qi.tar.Z
```

You may also see this referred to as "ph", which is what most of the clients are called. A collected set of clients for Macs, PCs, VMS, VM, etc, are in the

`/pub/ph.tar.Z` file.

There is also an archive of the mailing list for qi/ph software on the same machine. It's in /pub/info-ph.archive. You may join the list by sending email to info-ph-request@uxc.cso.uiuc.edu.

This software is supported by Paul Pomes <p-pomes@uiuc.edu> Contact him for more information.

Why can't I access the University of Minnesota's UPI news?

The University of Minnesota has a site license for UPI news, we are not allowed to distribute it off of our campus. We get our UPI news from Clarinet. For more information about getting UPI news send mail to info@clarinet.com. For information about setting up your own gopher-UPI server search the gopher-news archive for UPI.

When I do full-text searches I always get every document back, Why?

This is a problem occasionally encountered with Unix full-text indexes. It is caused by setting up the link incorrectly to a gindexd port.

The Path= field should be *blank* when pointing to a gindexd index.

Otherwise the client will send the path to the gindexd daemon, which interprets everything as a keyword. This path is likely to contain a pathname that is common to all of the indexed files. Thus a search generates hits on everything.

Where are there publicly available logins for Gopher?

Here is a short list, use the site closest to you to minimize network lag.

TABLE 7. Public Gopher Sites

Hostname	IP#	login as	area
consultant.micro.umn.edu	134.84.132.4	gopher	North America
pandaa.uiowa.edu	128.255.40.201	panda	North America
gopher.sunet.se	192.36.125.2	gopher	Europe
info.anu.edu.au	150.203.84.20	info	Australia
gopher.chalmers.se	129.16.221.40	gopher	Sweden
tolten.puc.cl	146.155.1.16	gopher	South America
ecnet.ec	157.100.45.2	gopher	Ecuador

We recommend running your own local gopher client though. The response time will be much faster, and you'll be able to save and print files.

How can I get my server published in the "Other Gophers" list?

When you have your server ready you can publish it to the world by sending e-mail to the maintainters of the "Other Gophers" list:

If your gopher server is in Europe, send mail to: gopher@ebone.net

Otherwise send mail to:

gopher@boombox.micro.umn.edu

CHAPTER 7

Common Questions

This chapter contains some of the more common questions that we've had the please to answer, and answer, and answer.

Overview

The most recent version of these "Common Questions can be gotten through gopher, or through anonymous ftp to the host:

```
rtfm.mit.edu
```

as

```
/pub/usenet/news.answers/gopher-faq
```

Those without FTP access should send e-mail to mail-server@rtfm.mit.edu with "send usenet/news.answers/finding-sources" in the body to find out how to do FTP by e-mail.

.

TABLE 6. Go4gw variables and routines

Variable/Routine	Function
$Ghost	Can be used by gateway routines to determine which host go4gw is running on.
GSERVER	File handle which is opened when GopenServer is called.
&Grepry(reply)	Sends string back to gopher client with \r\n on the end.
&Gabort(message)	Sends error message back to gopher client with "3mess\r\n.\r\n" on end.
&GopenServer(server,port)	Opens TCP port on server or calls &Gabort().
&GcloseServer	Closes server opened by &GopenServer()
$Gdebug	Controls debug messages in &Gsend() and &Grecv()
$Gport	Can be used by gateway routines to determine which port go4gw is running on.
&Gsend(cmd)	Sends "cmd\r\n" to GSERVER.
$_ = &Grecv;	Gets a response from GSERVER and strips \r and \n
&Gsorry	Sends a message about data that cannot be delivered due to restrictions.

Other "standard" routines and variables may be added. They will start with a 'G'.

Your `*_main` will be passed the string sent to go4gw WITHOUT your gateway prefix. For example, if someone sends the following string to go4gw:

```
nntp ls su.jobs
```

Then go4gw will call `&nntp_main("ls su.jobs")`, after loading g2nntp.

You should define any variables that users might have to change at the front of your script, and prefix them with your gateway name.

When your gateway has to return selector string, hostname, and port, it should use the following variables:

$Ggw -> name of this gateway

$Gport -> port this gateway is running on

$Ghost -> host this gateway is running on

For example, nntp might do the following:

&Greply("0$Subject\t$Ggw article $group $article\t$Ghost\t$Gport");

So when the user selects this the client will send:

```
nntp article su.jobs 104
```

Back to the go4gw daemon, which will figure out that "article su.jobs 104" needs to get passed to g2nntp.

The following variables and routines are defined in go4gw, and can be used by gateways:

TABLE 6. Go4gw variables and routines

Variable/Routine	Function
$GnotServer	You can define this in perl scripts that want to include the go4gw script without running the server. See the g2nntp_groups script.
$Ggw	Can be used by gateway routines to determine their gateway name.
$Gport	Can be used by gateway routines to determine which port go4gw is running on.

Why aren't these two auto-magically figured out? Mainly for speed, but also because some 'hostname' commands don't have the domain, some do, etc. Its easier just to define them here. Since all the other gateways are run in the context of this perl script, the gateways don't need any of this stuff in them.

`Gconf_file` should be set to the location of your `go4gw.conf` file. The format of this file is:

```
#
# format
# gateway : user : module : gopher title :
whois:-2:/usr/local/etc/g2whois:Whois:
nntp:-2:/usr/local/etc/g2nntp:USENET News:
webster:-2:/usr/local/etc/g2webster:Webster:
```

Where gateway is the name of the gateway, user is either a numeric uid or name, module is the name of the perl script that go4gw will dynamically load, and title is the title that will show up in the gopher menu if go4gw is sent the empty string "". If the gateway doesn't take an empty string, the title should be "" and it won't show up in the menu. By writing all your gateways so they take a "" command, you can point a link at the go4gw daemon with path set to "" and get a menu of all your gateways. The menu order will be the same order as the go4gw.conf file.

Writing New go4gw Gateways

go4gw gateways need to a follow a few simple conventions:

You need to have a routine called "${gateway}_main", where gateway is the name of your gateway. For example, if your gateway is called whois, then you need:

```
whois_main {
local($_) = @_;
 ...
}
```

In your module, for example `/usr/local/etc/g2whois.`

- NNTP - the USENET news transport protocol.
- Webster - the Webster dictionary service available on some NeXT machines.
- Whois - services that provide information about users and organizations.

Requirements

You will need a UNIX machine on a network with a working version of perl. The source code for Perl can be gotten from various FTP sites on the Internet. The canonical location of perl is the anonymous ftp site `prep.ai.mit.edu`, as

```
perl-4.036.tar.z
```

You will, of course, need the go4gw software distribution. See *"Getting the Internet Gopher Software"* on page 9 if you don't have the go4gw software.

Installation

Copy all of the perl scripts that comprise go4gw into a common directory, in our examples we'll use `/usr/local/etc/go4gw` as our directory.

You need to put the following line in `/etc/services`:

```
go4gw 4320/tcp
```

And the following line in `/etc/inetd.conf` (depending on your system type):

```
go4gw stream tcp nowait /usr/local/etc/go4gw go4gw
```

The go4gw script has a few variables you might want to change:

```
$Gconf_file = "/usr/local/etc/go4gw.conf";
$Gport=4320;
$Ghost="Slapshot.Stanford.EDU";
```

You need to set `Gport` to the same port as in `/etc/services`, and `Ghost` to your fully qualified host name.

CHAPTER 6

Go4gw - The Gopher Gateway

Go4gw is a program that provides gateways to many popular internet services, such as USENET news, whois, and archie. The following chapter describes go4gw and how to install and manage it.

Overview

There is quite a bit of information that could be in gopher, but isn't. The use of a Gopher Gateway makes it possible to include this information inside of Gopherspace.

Go4gw is designed to channel these other forms of information into a format suitable for gopher. It is written in perl and allows easy extensibility, a number of prewritten gateways and easy administration.

Gateways to the following services are provided with go4gw1.01

- Archie - a database of files available for anonymous ftp.
- Finger - a service that provides information about people logged in to UNIX or VMS hosts.
- Geo - the geographic name server at the University of Michigan.

If you want to install everything in the directories specified in Makefile.config type:

```
make install
```

If you want to install just the client type:

```
(cd gopher; make install)
```

If you want to install just the server type:

```
(cd gopherd; make install)
```

Testing

....

DEC Ultrix Systems

Change the line that contains -lcurses to have -lcursesX instead.

Other Systems

There is a line in the file Makefile.config for compatibility definitions. Look for the line with COMPAT= at the beginning. If you have problems linking due to unresolved symbols you'll want to edit this line. Consult the following table for the appropriate definitions you should add if you have problems.

TABLE 5. Compatibility Definitions

Unknown symbol	What you should add to COMPAT=
strstr	-DNOSTRSTR
tempnam	-DNO_TEMPNAM
strdup	-DNO_STRDUP
bzero or bcopy	-DNO_BZERO
getwd	-DNO_GETWD
tzset	-DNO_TZSET
strcasecmp	-DNO_STRCASECMP

You should then type **`make clean`** and restart the compilation process.

Compiling and Installing the Distribution

You should now be ready to compile the distribution.

If you want to make just the client type

```
make client
```

If you want to make just the server type

```
make server
```

If you want to make everything just type

```
make
```

DL Database Support

Add `-DDL` and define `DLPATH` and `DLOBJS` if you want to include support for Tim Cook's 'dl' databases You will also have to have a working copy the program with source code in the `DLPATH` directory. The files `getdesc.o` and `enddesc.o` must be there.

Source for dl can be gotten from:

```
admin.viccol.edu.au
```

as

`/pub/dl/describe-1.8.tar.Z` or higher...

Special Modifications for Compatibility

The Gopher distribution tries to automatically make up for operating system dependencies, however sometimes you have to tweak things for certain operating systems. Follow the appropriate steps if your system is listed below:

UMAX Systems

Uncomment the line that contains UMAXLIBS in Makefile.config

Dynix PTX Systems

Uncomment the line that contains PTXLIBS in Makefile.config.

Dynix systems in the BSD universe

Uncomment the line that contains SEQLIBS in Makefile.config.

SCO Open Desktop Systems

Uncomment out the line that contains SCOLIBS in Makefile.config.

System VR4 Systems

Uncomment out the line that contains SVR4LIBS in Makefile.config.

TABLE 4. Configuration Parameters in conf.h

Parameter	Default	What it does
PLAY_COMMAND		Commaand used to play sounds.
MAIL_COMMAND	mail	Command used to mail files.
TELNET_COMMAND	telnet	Command used to connect to other hosts using telnet protocol.
TN3270_COMMAND	tn3270	Command used to connect to IBM mainframe hosts.
IMAGE_COMMAND	xloadimage -fork %s	Command used to view image files.

Other Optional Features

Many people have submitted nifty options for the server, follow the instructions in the appropriate section if you wish to implement these features.

Load Restriction

Add `-DLOADRESTRICT` to the `SERVEROPTS` variable in `Makefile.config` if you want to restrict access to the Gopher server at a certain load average. You will also need to add -lkvm to `SERVERLIBS` in Makefile.config.

This has only been tested underneath SunOS 4.1.1

Biological Databases

Add `-DBIO` to `SERVEROPTS` if you're using the biology portion of Don Gilbert's modified wais8b5 that supports boolean and phrase searching. PLEASE NOTE: the `-DBIO` option is NOT needed nor recommended for use of the boolean and phrase searching portion of this modified wais, just the symbol searching. But, you must compile wais and gopher with the same option setting (`-DBIO` or NOT). Otherwise your server will core dump when doing searches.

Two files control the compilation process: `Makefile.config` and `conf.h`. You will find both of them in the top level of the Gopher source directory. Most sites will not need to make major changes to these files, in fact you might just want to try typing "make" and see if it works for you.

There are comments in these files that you will want to read before you compile. The following tables document some of the more popular ones.

TABLE 3. Configuration Parameters in Makefile.config

Parameter	Default	What it is
CC	cc	Your favorite C compiler
OPT	-O	Use -O for faster code, -g for debuigging
RANLIB	ranlib	Use touch if you don't have the ranlib command.
PREFIX	/usr/local	The base directory where the software will be installed. Many other parameters use this value.
CLIENTDIR	$(PREFIX))/bin	Where the client "gopher" is installed.
CLIENTLIB	$(PREFIX)/lib	Where the help filesforthe client are installed.
SERVERDIR	$(PREFIX)/etc	Where the server files are installed.
SERVEROPTS		Optional server features. See below.
DOMAIN	.micro.umn.edu	Set this to the portion of your hostname that the "hostname" command doesn't return.
SERVERDATA	/home/gopher	The defauilt location of gohper data for the gopher server.
SERVERPORT	70	The default port for the Gopher server.

TABLE 4. Configuration Parameters in conf.h

Parameter	Default	What it does
CLIENT1_HOST	gopher.tc.umn.edu	The default host to connect to.
CLIENT2_HOST	gopher2.tc.umn.edu	The alternate host to connect to.
CLIENT1_PORT	70	The default port for the first host.
CLIENT2_PORT	70	The default port for the alternate host.
PAGER_COMMAND	more -d %s	Command used to display text.

The Don Gilbert Version supports boolean searches and phrase matching. The normal WAIS software does not support these features.

To retrieve the Thinking Machines Inc. version of WAIS, use anonymous ftp to the host `ftp.think.com` and retrieve the file `/wais/wais-8-b5.1.tar.Z`

To retrieve Don Gilbert's version of WAIS use anonymous ftp to the host `ftp.bio.indiana.edu` and retrieve the file `/util/iubio-wais-8b5.tar.Z`

The source code to the WAIS system is in this compressed tar file. You should then extract the archive and compile the software. The installation instructions are in the file INSTALLATION inside the wais doc directory.

Note that you do not need to make the Xwindow or emacs client. Many systems will not compile these cleanly.

When you are finished you will have a bin directory that contains the programs: `waisindex`, `waisserver`, and `waissearch`. These binaries allow you to create, serve and search indices respectively.

The bin directory contains the libraries necessary for compiling the gopher server. These are `inv.a`, `libftw.a`, `client.a`, and `wais.a`. The ui directory contains one module necessary for the WAIS gateway, `source.o`.

Go to your Gopher source code directory and type the following commands to let the Gopher code know you want to use WAIS. Substitute the WAIS source code directory for *waisdir* in the following commands.

```
ln -s /waisdir/ir .
ln -s /waisdir/bin .
ln -s /waisdir/ui .
```

Preparing to Compile the Distribution

There are some options you will want to check before compiling the Gopher server, some of these options are for compatibility with various platforms, others are for special features.

- Apples running A/UX
- DECstations running Ultrix
- Silicon Graphics Iris' running Irix

Don't worry if your system isn't listed above, the code has been written with compatibility in mind. Most systems should work just fine.

NeXT Indexing

For NeXT indexing you will need the file `NeXTtext.tar.Z`. Use anonymous ftp to `boombox.micro.umn.edu` and get the file

```
/pub/gopher/Unix/NeXTtext.tar.Z
```

The files `libbtree.a` and `libtext.a` are in this compressed tar file, along with the directories `btree` and `text`.

The files `libbtree.a` and `libtext.a` need to be installed in the directory

```
/usr/local/lib
```

The `btree` and `text` directories need to be installed in `the directory`

```
/usr/local/include.
```

Type the following commands as root to install the NeXT indexing libraries.

1. **`uncompress NeXTtext.tar.Z`**
2. **`tar xvf NeXTtext.tar.Z`**
3. **`mv libtext.a libbtree.a /usr/local/lib`**
4. **`ranlib /usr/local/lib/libbtree.a /usr/local/lib/libtext.a`**
5. **`cp -r text btree /usr/local/include`**

WAIS Indexing

For WAIS indexing you will need to get the freely available WAIS software. There are two different versions of WAIS software: the software from Thinking Machines Inc. and the modified software from Don Gilbert at the University of Indiana.

You will need a Unix machine with a C compiler and development environment to compile the software. The source code requires about one megabyte of disk space when uncompressed. The source takes up about two megabytes of disk space when compiled.

Your UNIX machine will need to be connected to a TCP-IP network for the software to function properly.

Some experience with Unix software installation is recommended. A knowledge of C programming and Makefiles will be useful.

If you want to index files with your server you will need to get some additional software. There are two file indexing kits supported by the Internet Gopher server, NeXT indices and WAIS indices.

NeXT indexing comes bundled with the NeXTstep operating system. It is the basis for many NeXT applications, including the Webster dictionary and the Digital Bookshelf.

WAIS indexing is a freely available indexing kit for many Unix machines. It includes an index generator called *waisindex*, a database server called waisserver, and client programs for Xwindow, full-screen and command line searching. WAIS is an acronym for Wide Area Information Server.

If you are compiling a server on a NeXT machine you will want to consult *"NeXT Indexing"* on page 37 before continuing. Follow the instructions in *"WAIS Indexing"* on page 37 if you want to use WAIS indexing with your server. You may skip these sections if you don't want indexing or don't want to bother with it just yet.

Compatibility

The Gopher source should compile on most UNIX systems. Here are a few that we know it has worked on:

- Suns running SunOS 4.1.x
- RS/6000s running AIX 3.2
- NeXTs running NeXTstep
- SCO ODT systems

CHAPTER 5

Installing the Unix Gopher Distribution

This chapter describes how to compile and install the Internet Gopher for Unix Distribution. The two main parts of this distribution are the Gopher Server, *gopherd* and the Gopher client, *gopher.*

Overview

Internet Gopher for Unix is distributed as C source code. You will need to compile this source code into binary executables for your system.

To use the gopher client or server you will need to compile the code using a C compiler. To make it generally available to the rest of the users on the system, you should install the software into a central directory.

Requirements

You should have a copy of a compressed tar file containing the source. You should consult *"Getting the Software"* on page 9 if you don't have this file.

Type in a meaningful name. Later on, when you wish to access this item again, press the ‘v’ key to view your personal bookmark list. You’ll get a directory listing screen that contains all of your bookmarks. If you wish to delete a bookmark entry you can press the ‘d’ key when the arrow is pointed to the item.

Saving and Downloading

If you want a permanent copy of an item in Gopherspace, not just a reference, which a bookmark is, you should press the ‘s’ key. This will save the current item into a filename.

The following table summarizes the other keys:

TABLE 2. UNIX Gopher Client-Advanced Navigation Keys

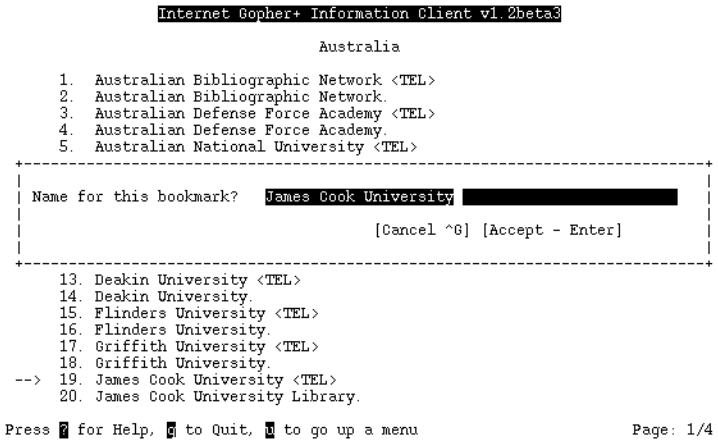
Key	Action
>, +, SPACE	Move to the next directory page.
<, -, b	Move to the previous directory page.
0-9	View a specific item.
m	Go back to the first “main” directory screen.

Bookmarks

If you formulate a search or find a folder you would like to come back to later, you can set a bookmark for the item. Then you can easily find the item later in your personal bookmark directory.

Move the arrow to the interesting item and press the ‘a’ key to add a bookmark. You will see a screen like this:

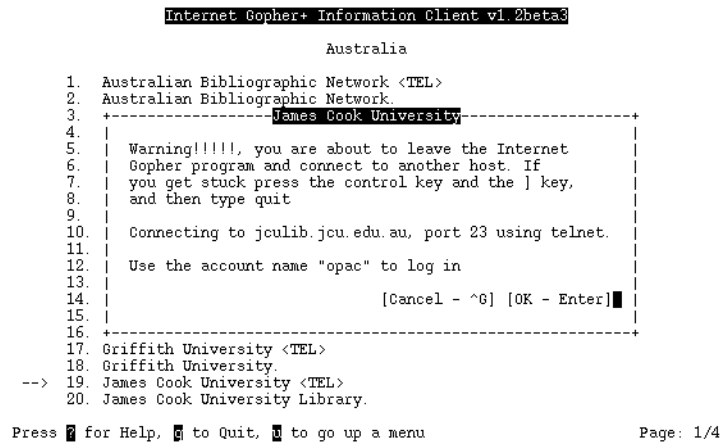
FIGURE 8. Unix Gopher Client-Adding a Bookmark



Telnet/tn3270 Servers

Gopher can connect to telnet sites such as Library Catalogs rather easily, if you see an item tagged with <TEL> or <3270> you'll be presented with a screen similar to the following

FIGURE 7. Unix Gopher Client-Telnet/tn3270 services



In this example we're connecting to the "James Cook University" library system. If there's a special login required Goppher will tell us. In this case it's "opac" If we press RETURN, Gopher will connect us to the host jculib.jcu.edu.au and let us login.

Advanced Navigation Features

There are many other keys that you can use with the UNIX client. Instead of the cursor keys you can use the the equivilant cursor control keys for vi or emacs. In vi the keys are h,j,k, and l. In emacs the keys are ^B, ^N, ^P, and ^F.

Special Items

This section details how to use the Unix Gopher Client to access services other than text files and directories.

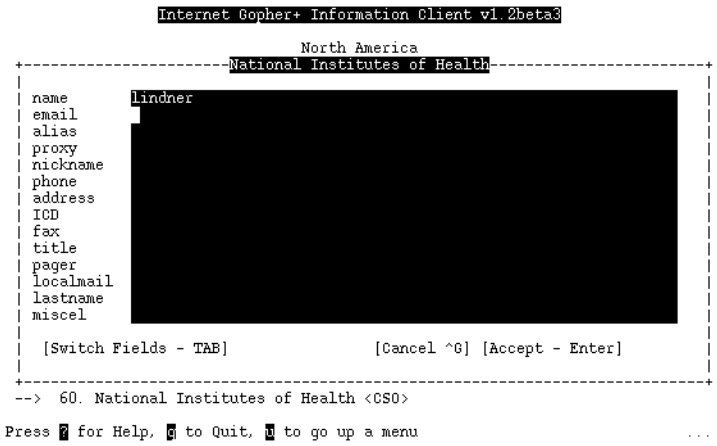
Index Searches

Items with a <?> tag at the end of the name are searchable indices. When you select one of these items you will get to type in key words. These words will determine the items in the resulting directory listing. For instance a search on "*Search Recipes*" with word *salmon* would result in a listing of recipes related to salmon.

Phone Book Servers

Phone Book Servers show up with the tag <CSO> at the end of the name. Selecting one of these items results in a screen that looks like this:

FIGURE 6. Unix Gopher Client-Phone Book Search



You can fill in the fields in any way you wish. In this example we want to see if there are any people with the name "lindner" at the National Institues of Health.

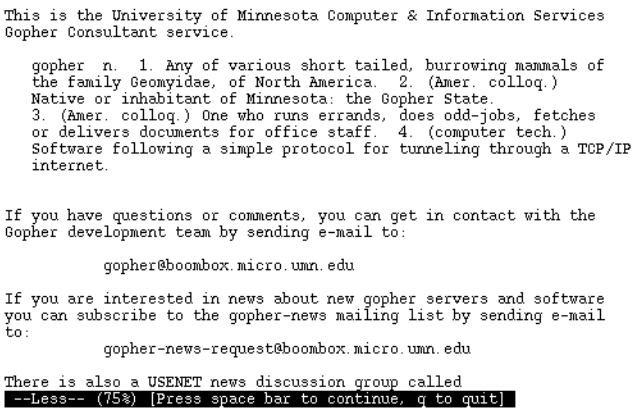
The cursor keys control the arrow pointer, pressing the cursor down key moves the arrow to the next item, pressing the cursor up key moves the arrow to the previous item.

Pressing the RETURN key or right arrow key displays the item that the arrow is pointed at.

Each item is tagged with special characters to make it easy to predict what each item is. Items ending with a slash (/) are Gopher directories, selecting one of these items will cause a new directory to be displayed.

An item that ends with a period (.) are text files. Pressing RETURN on these items will view the file. For instance pressing RETURN on the file “About Gopher” will retrieve the file and result in a screen that looks like the following:

FIGURE 5. Unix Gopher Client-Text File



Press the ‘q’ key when you’re finished viewing a file to return to the directory

Once you’ve finished with viewing the directory you can, if you wish, return to the previous directory by pressing the ‘u’ key or the left arrow key. This moves you *up* to the previous directory.

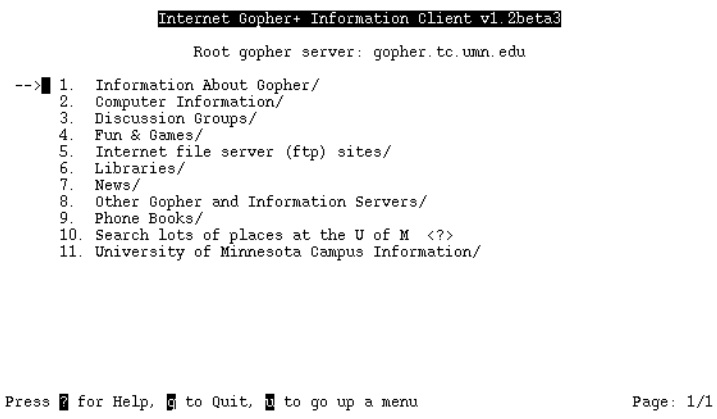
These simple keys will move you through Gopherspace quickly and easily.

If you get an error message, then you will have to install the UNIX Gopher client, see *"Installing the Unix Gopher Distribution"* on page 35, or you'll have to use one of the public gopher systems listed in the Common Questions chapter.

Starting the UNIX Gopher Client

To start the UNIX Gopher client, simply type **`gopher`**. You'll be presented with a screen that looks like this:

FIGURE 4. Unix Gopher Client-Top Level



This is what you would see if you used gopher at the University of Minnesota, other systems may have localized and customized the client for their own sites.

Navigating Gopherspace

Once you've started the client you can move around Gopherspace using a few differnt keys, the most common ones are the cursor key, the u key, and the RETURN key.

CHAPTER 4

The UNIX Gopher Client

This chapter describes how to use the UNIX Gopher client.

Overview

Internet Gopher for Unix is distributed as C source code. You will need to compile this source code into binary executables for your system.

To use the gopher client or server you will need to compile the code using a C compiler. To make it generally available to the rest of the users on the system, you should install the software into a central directory.

Requirements

You will need a working gopher client, to see if it's been installed just type:

```
gopher
```

The TurboGopher Settings file

We don't want you mucking around inside the data fork of a Settings file, so we aren't going to encourage you to do it by spelling out the format (though anyone with a disk editor and a room temperature IQ can figure it out).

We DO want you mucking around inside the resource fork of the Settings file though. When it is created by TurboGopher, the TurboGopher Settings file has no resource fork. If you add one by opening the file with ResEdit, then TurboGopher will place the Settings file in its resource search path. This means that if you want some custom resource to be used by TurboGopher, you may place it in the resource fork of the Settings file rather than munging around inside TurboGopher's application resource fork.

We would like you to place things like string localizations, custom code resources, and overridden configuration resources here for your local users. This way, you can distribute a localized TurboGopher Settings file on your campus and you shouldn't have to scramble to patch the application every time we release a new version. Certainly we might add more strings to various STR# resources... but if you drop us a line and tell us what you are doing with the resources, we will inform you before we release any new versions of TurboGopher that might affect you.

We have made provision for TurboGopher to execute external code resources to do certain tasks. At this point we're deciding if we should adopt some new interface for these code resources or to use the interface that Mac folks are very familiar with: XCMDs. We will probably wind up giving you access to various internal parts of TurboGopher via XCMDs.

'BookMarks'. It contains Gopher item descriptors EXACTLY as they come in off the net, ie:

Display-String<TAB>Selector-String<TAB>hostName<TAB>port<CR><LF>

NOTE: Ugly linefeed after the <CR>

The built-in manual is stored as 'TEXT' resource called "HeLp". While you can replace that, we certainly don't encourage you to do so.

Finally, most of the important English text for status messages etc. are stored in STR# resources, The primary one of these is STR# resource ID 3003. This is important for international localization purposes.... but read on.

Bookmark file formats

Bookmark files contain just the Gopher item descriptor lines that a Gopher client would ordinarily retrieve over the net. Please read the Internet Gopher protocol document for more information on this. Gopher bookmark files contain text. The fileType is not 'TEXT' however, so if you wish to edit one with a word processor or text editor, you will have to either change its fileType or use a permissive editor. Each line contains the familiar:

Display-String<TAB>Selector-String<TAB>hostName<TAB>port<CR>

NOTE: This is Mac text... SO NO LINEFEEDS AFTER THE <CR>

Bookmark files have the type 'BKMK' and creator 'MDDF'. The Import Bookmarks menu item allows you to ATTEMPT to read any filetype you want and treat it as a Bookmark file. Please use this with caution. While TurboGopher will attempt to make sense out of the file, and will reject most junk files, you definitely are on thin ice here. The Open Gopher Bookmark File menu item however will only let you open bonafide Bookmark Files.

ISO Translation Table

The ISO Latin-1 character set translation tables used by TurboGopher were kindly supplied by Jean-Pierre Kuypers <Kuypers@sri.ucl.ac.be>, Pascal Maes <Maes@elec.ucl.ac.be>, and Andre Pirard <Pirard@VM1.ulg.ac.be> They are stored in taBL resources with ID 1001 and 1002.

If the user had deleted all bookmarks, the Bookmarks window is not opened (the contents of the Bookmark worksheet window are remembered via the Settings file).

Next TurboGopher looks to see if the user has launched it by double-clicking on a Bookmark file (or by dropping a Bookmark file on it under System 7). If this is so, the contents of the Bookmark file (more on format later) are read and placed in a list (directory) window; in this case TurboGopher will not make a connection immediately to the Home Gopher Server.

If TurboGopher was launched directly (not via a Bookmarks document) then it connects to the Home Gopher Server, retrieves the contents of its top directory, and places this in a list (directory) window. At this point, TurboGopher is fully up and running. Next we need to describe the various parts alluded to above.

The Home Gopher Servers

Out of the box, TurboGopher comes preconfigured to connect AT RANDOM to one of the two primary gopher clones (gopher.tc.umn.edu or gopher2.tc.umn.edu; the former is also known to old Gopher folks as gopher.micro.umn.edu) at the University of Minnesota. It must be emphasized that these are NOT a primary and secondary server: they contain exactly the same information and changes made to one are automatically propagated to the other. If the first connection attempt fails, TurboGopher will try the next. This has two purposes. First, initial connections are very reliable for users and since we serve as the world's Gophermeisters, this is good. More importantly for us, the randomization ensures that users will spread the load over our two small, economical, primary servers (Mac IIci's). If your campus' primary Gopher server experiences heavy use, we suggest you make a clone too.

The user may of course reconfigure for only one primary server. TurboGopher remembers this via the Settings file. A campus network administrator may directly set TurboGopher's internal primary server defaults by altering STR# ID 3002 using ResEdit. The first 3 strings are for one alternate server, the next 3 are for the other. Leaving one or the other triplet empty implies that no clone servers are available. The first string in the triplet is the server's domain name, the second is the port, the third denotes whether it is a Gopher+ server (ignore this for now).

Other Built-in Information

A campus network administrator may also change the default bookmark set that TurboGopher carries around with him. This is stored in a 'TEXT' resource called

changes the helper application that TurboGopher asks the finder to launch to view the document. Double-click the view in the list and you will be presented with a dialog in which you can select the application to be used as a TurboGopher helper.

By the way, you won't be able to do this if you're running System 6...

Miscellaneous

TurboGopher saves or caches the item lists for all visited directories (even when windows are closed). Since these lists don't ever have to be fetched from the server again, this dramatically increases speed when choosing any item from the Recent menu. If TurboGopher runs out of memory it will first inform you and then automatically release the caches for closed windows.

As mentioned earlier, you may selectively release a directory's cache by closing the window while holding down the Shift key if you wish more control over memory.

TurboGopher stores what it needs in a file called TurboGopher Settings; this lives in the Preferences folder, which is found in your System Folder. If you drag this file out from the Preferences folder and place it in the same folder as TurboGopher, it will still be found and used.

For Experts

This section describes features of TurboGopher that either network administrators or Gopher weenies would want to know.

Waking up TurboGopher.

When TurboGopher starts up, it looks for the TurboGopher Settings file first in its own folder, and failing that in the Preferences folder. If it finds the TurboGopher Settings file, it uses the settings and resources (more on this later) it finds inside. If it cannot find the file, it creates a new one in the Preferences folder based on its internal defaults (more on this later too). If there are any bookmarks the user had set in the Bookmarks worksheet window, the Bookmark worksheet window is opened.

pher will randomly try one of the two and then try the other if the first is down. If you don't understand this, you don't have to bother with it or with the Another Gopher item under the File menu.

TurboGopher attempts to speak Gopher+ to all Home Gopher servers, and then backs off if it sees that the server does not have Gopher+ style items. This is a bit presumptuous, and it confuses some servers. These servers should be revised in the future so they don't confuse so easily. If configured to connect to a Home Gopher server and nothing is displayed, you may want to try using the 'Another Gopher' item under the file menu and making sure that the gopher+ check box is NOT checked.

TurboGopher Options

You can use the Options item under the Setup menu to customize some behaviors of TurboGopher. The Single Directory Window checkbox makes TurboGopher recycle its windows whenever you open a new directory (and not just when you hold down the Option key).

Use the ISO Latin-1 checkbox if you need to use Latin/Romance language characters. Many sites in Europe have a need for this to display their national characters. If you visit European gopher servers and the accented characters don't display correctly, you may want to choose this option.

The Extended Directory listings checkbox is disabled; it's for a future feature.

If you visit Gopher servers with Japanese or Chinese documents, and you use a Japanese or Chinese font while running System 7.1's WorldScript, Gopher will display Japanese SJIS (MS Kanji) or traditional Chinese characters (BIG-5).

When you save documents, TurboGopher normally saves them as MacWrite II TEXT documents (we like MacWrite II). If you'd rather save them as TEXT documents of your favorite word processor or editor, use the appropriate 'Set' button in the Options dialog to select the word processor of your choice.

With Gopher+ servers, items may be available in a number of alternate views. When fetched, these alternate views are typically saved as a file to be viewed by a Mac application. If you are a network/macintosh cowboy, you can change the Mac application and Mac filetype associated with a Gopher+ alternate view.... this

Use the ‘Get Attribute Info’ menu item of the Gopher menu for this.

Canceling Slow Network Operations

TurboGopher executes most requests in a few seconds. A document or directory is displayed as it is received. However, if a server is especially slow or busy or if you have started to fetch what appears to be a very long and uninteresting document or directory, you may cancel the fetching process by closing the document or directory window. Once you have closed the window, TurboGopher assumes that it does not need to fetch the remainder of the item. Note that when you fetch a file that is saved to your Mac disk, TurboGopher displays a window with the status of the fetch. Again, to cancel the fetch operation you can simply close the window.

A more extreme method for canceling a network operation is to choose the Cancel All Requests item from the Gopher menu (or by holding down the command key and typing a period). If you open a file or folder by double-clicking while holding down the Option key (to conserve screen space as described earlier), TurboGopher will also cancel the running network transaction if one is outstanding; and this is in fact exactly what you want over low speed connections such as SLIP. Both these features make TurboGopher seem faster and more responsive over SLIP links.

TurboGopher runs happily in the background (if you wish to fetch something very large or from a very slow server). You may also fetch an item (file or directory) while another one is still being fetched, since TurboGopher supports multiple concurrent streams. Command-Period will cancel all active streams.

Starting and Configuring TurboGopher

The ‘Start Gopher’ item under the File menu opens a window containing the initial directory fetched by connecting to your home Gopher server. TurboGopher is preconfigured to connect on startup to one of the University of Minnesota’s Gopher servers. You may change your home Gopher server by using the ‘Configure’ item under the Setup menu. To do this you need the full Internet domain name and port of the desired primary server. You can configure TurboGopher to use one of two alternate primary servers. This distributes the load for a campus over two different (but equivalent) primary servers and it prevents a single point of failure. TurboGo-

ticket for each request. This means that your password is never sent over the network, and that you only have to enter your password to get the initial authorization.

Bookmarks and Bookmark files

If you formulate a search or find a folder you would like to come back to quickly later, you can use the Set Bookmark menu item to save your place. Set Bookmark saves a reference to a folder, file, search, terminal session... in fact any Gopher item. If nothing is selected in the front list window, a bookmark is made for a folder representing the window. Try it, it's easier to do than to say.

Bookmarks are placed in a special Bookmarks window. Use the Show Bookmarks menu item to open the Bookmarks window. Bookmarks are remembered even after you quit TurboGopher; contents of the Recent menu are however forgotten. It might be useful to consider the Bookmarks window to be a special kind of 'worksheet' or 'construction area' for you to build your own scratch-pad of useful gopher resources.

You can save your Bookmarks window or ANY list window or ANY selected item as a Bookmark file ('Save as Bookmark FileI' menu item). Bookmark files are small and may be exchanged with other Gopher users if you wish.

The 'Import Bookmarks' menu item lets you read a Bookmark file right into your Bookmarks window. The 'Open Gopher Bookmark File' menu item on the other hand, places the contents of a bookmark file in a window of its own. The latter is equivalent to double-clicking on bookmark files from the Finder. If you launch TurboGopher by double-clicking on a bookmark file, it will not immediately connect to the Home Gopher server, but will open the Bookmark file instead. So you may use bookmark files as custom Home Gopher servers.

Item Attributes

More sophisticated Gopher servers (referred to as Gopher+ servers) can provide you with information about any selected item, much like the Finder's Get Info menu lets you get information about a Mac item.

The Recent menu lets you go back to any directory window you have viewed during your Gopher session. The left and right arrow keys may be used to go to previous or next windows; they are equivalent to traversing the recent menu upwards or downwards. Note that the Recent menu only remembers directory windows you visit. Document or phone-book lookup windows are not listed.

If you close a window by clicking in its close box while holding down the Option key, then (like Finder) TurboGopher will close all open windows.

If you click in the close box while holding down the Shift key, then the cache of information about that window will be released. So the next time you open this window from the recent menu, TurboGopher will have to fetch the contents over the net once more.

Of course, if you hold down both Option and Shift keys then all windows are closed and all caches released. Caches are explained below.

For the terminally curious, if you hold down the Shift key while single clicking on an item in a list, TurboGopher will display the item's Gopher selector string, host name, port number and Gopher+ baggage in the status pane. This is a quick peek. If you want to copy-paste this information, use the Get Attribute Info menu item found under the Gopher menu; the information you want is in the +INFO attribute (the first line).

If wish to fetch an item that is a Macintosh file (binhexed file, shows up with the icon of a disk), but don't want TurboGopher to automatically dehex it as it fetches, you may force TurboGopher to fetch the document as a raw, unprocessed file. To do this, hold down the Control key while you double-click on the item. For folks using DownLine or other applications for de-binhexing and de-archival, this is one way to do what you want.

Authenticated Servers

TurboGopher now supports AdmitOne Authentication for Gopher+ servers that restrict access to information on a per-user basis. When you attempt to open a directory that has such access restrictions, TurboGopher will prompt you for your username and password on the restricted-access server. TurboGopher and the server negotiate to establish a valid ticket, and subsequent requests to the server use a new

to work, you need to have Telnet or TN3270 installed on your Macintosh. If you are still running System 6, TurboGopher will only save a Telnet session file rather than launch Telnet; this is another good reason to run System 7 on your Mac.

Following the Macintosh guidelines, TurboGopher starts out displaying text in the standard application font. The Font and Size menu items let you change font appearance for documents and lists. TurboGopher will attempt to display phonebook lookup results in a monospaced font (Monaco), effectively ignoring your font selection. It IS possible to change the font of a phonebook results lookup window after it has been opened; however things will not line up correctly and generally look icky.

Advanced Navigation Features

If you double-click on a folder while holding down the Option key, TurboGopher will re-use the current window instead of using a new one to display the folder contents. Some folks like to do this to prevent screen clutter and keep from having too many open windows.

You may use the up and down arrow keys on your keyboard to move up and down lists in windows. The Enter or Return key opens an item (and is equivalent to double-clicking on an item). Use of the arrow keys is not supported on the Mac Plus.... (sorry).

You may also type the first letter (or first few letters) of an item name in a list and TurboGopher will highlight the first item it finds that matches (sort of like the Finder does in list views). The Find menu item is available to locate an item in a list: just type a string in the Find dialog. Find can also be used to search for text in open document windows, by the way.

To delete any item in any list window, click once on the item to select it. Then choose ‘Delete Item’ from the Gopher menu.

TurboGopher remembers the last full-text index search (item with question-mark icon) you used. If now you hold down the Option key and double-click on a word in a document window, TurboGopher will query that search service for all documents containing the word you selected. You must use a search service before this pseudo-hypertext capability is accessible (TurboGopher needs to know which search service to use in your hypertext query).

Items that are specific to MS-DOS are seen as documents with a tiny 'PC' emblazoned on them; similarly items specific to UNIX show up as documents with a tiny 'UX' on them. You can retrieve either of these kinds of items if you like. TurboGopher will ask you if you wish to save them to your disk. You may not 'view' these items as text. In fact it will probably make very little sense to fetch such items unless you intend to further transfer them later from your Mac to another kind of computer, or if you have some special tools on your Mac that will allow you to use these kinds of items.

Pictures

Items appearing as documents with a starburst on them are probably graphics or picture files (typically in GIF, JPEG, or PICT format). TurboGopher will transfer such files, save them on your disk, and optionally launch a picture-viewing application if you wish to view the picture.

Alternate Views

Gopher+ server may store more than one representation of a document (an image, text, sound, video, etc.). If more than one view of a document is available, you can select between the views by choosing the 'Get Alternate Views' item from the Gopher menu. When alternate views are available, an alternate view window will be displayed with descriptions of the views.

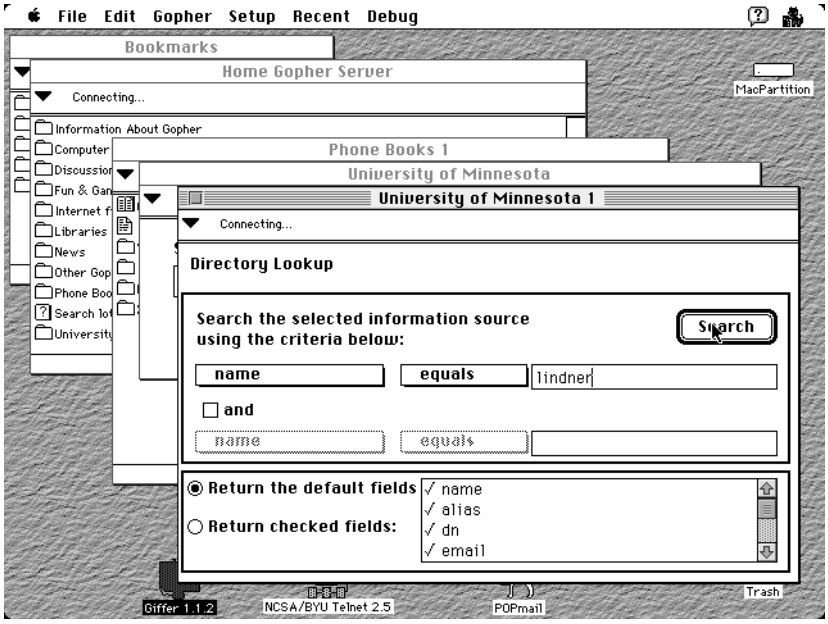
For many alternate views of documents, TurboGopher will save a copy of the item to your Macintosh's disk, and then (under System 7) ask the Finder to launch a helper application to actually display the item. You can configure which applications should be used as TurboGopher helpers; to do this see the section on configuration options.

Connecting to Terminal-based Services

Opening the 'terminal' icons will make TurboGopher launch NCSA Telnet (or TN3270 if appropriate) and start a terminal session to a terminal-based information system. Typically these are library catalogs or other such services. Note that for this

name field, an address field, and a phone number field). The simple phone-book lookup dialog assumes that you want to search for someone by name. Click on the 'More Choices' button if you wish to specify exactly which fields to search in, when you formulate a query. The result of the query is displayed in a document.

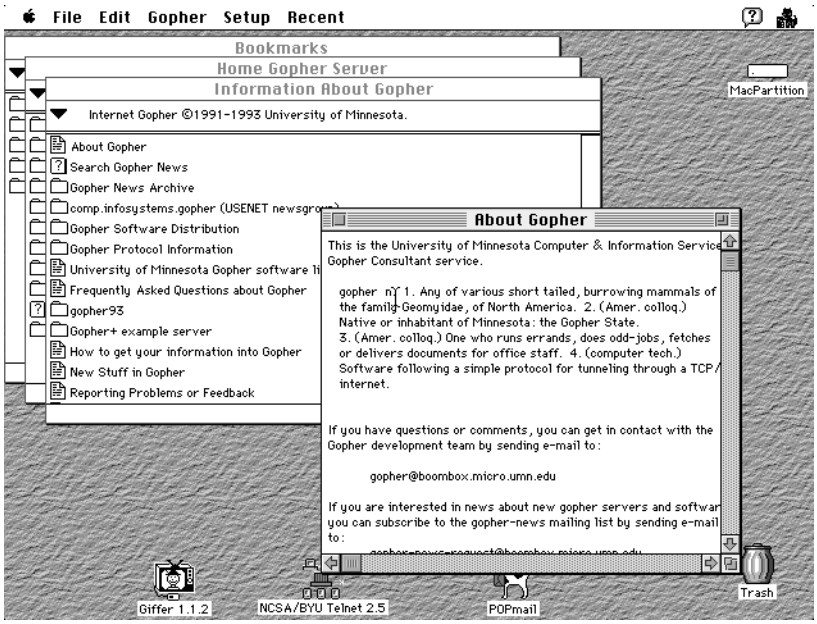
FIGURE 3. TurboGopher- Phone Book Search



Software Archives

Disk icons represent archived Mac software or documents that you can fetch; these items will be saved to your hard disk. Such items are often stored in compressed form to save space. You need to have certain Mac utilities to 'uncompress' such items. The most common such utilities are StuffIt and Compactor. Both may be obtained from the Info-Mac archives with gopher. If you are running System 7 and TurboGopher retrieves an item that needs to be 'unstuffed' or 'uncompacted', it will ask you if it should open them via Finder and the appropriate decompress utility. We recommend that you use CptExpand and SitExpand for your decompress needs; both of these are available with the TurboGopher software distribution as well as from the Info-Mac archives.

FIGURE 2. TurboGopher-Top Level



Navigating Gopherspace

To navigate through gopherspace, double-click on any interesting items to open (fetch) them. If you double-click on a file, TurboGopher will fetch and display the file. Opening a folder will let you view its contents. Double-clicking on the question-mark (Search) icons will let you search a database. On most gopher servers these databases are full-text indexes of a collection of information. Full-text index means that every word in every document is considered a keyword. The best way to search a gopher full-text database is specify the words for which to search when TurboGopher presents you with a dialog box. The results of the search are returned as a list of documents containing those words.

Phone Books

If you double-click on the phone-book icons you can search electronic phone books. Phone book databases are structured in the sense that they have fields (i.e. a

ing information as soon as possible... you can read the first part of a document or directory while the rest is being fetched. This version of TurboGopher also supports many of the Gopher+ extensions to the original Internet Gopher protocol.

In spite of the design goal to run fast as possible, TurboGopher is a good Mac citizen: it shares time with other applications. You can put TurboGopher in the background to fetch lengthy items in the background while you work in another application in the foreground.

Requirements

You will need a copy of the TurboGopher application, if you haven't retrieved this yet refer to *"Getting the Internet Gopher Software"* on page 9. You will also want to retrieve the "helpler applications." These applications will allow you to connect to telnet sites, view graphics, etc.

TurboGopher requires a Macintosh running a system 6.0.7 or later. We highly reccomend using System 7 or higher. Without it you won't be able to automatically launch the "helper applications."

You should have at least one megabyte of memory, more if you're using Multifinder or System 7.

You will need a correctly configured copy of MacTCP and a network connection. If you don't have MacTCP you can get it from the Apple Programmers and Developers Association (APDA). MacTCP is a licensed product, it is not free.

Getting Help

The information in this chapter can also be found by picking the menu item "Help" from the "Gopher" menu of TurboGopher.

In addition to this information you are reading right now, if you are using System 7 (and we strongly recommend that you do), some Balloon Help is available in TurboGopher. See your Macintosh System 7 documentation for information on using Balloon Help.

CHAPTER 3

TurboGopher

TurboGopher is a Gopher client for the Macintosh. The following chapter contains information on using, configuring and administrating the TurboGopher application.

A Websters definition is:

TurboGopher n. 1. A small rodent with a turbocharger strapped on its back to increase its speed and ferocity. 2. (Amer. colloq.) Native or inhabitant of Minnesota after consuming three double espressos. 3. (Amer. colloq.) An Olympic sprinter who runs errands, does odd-jobs, fetches or delivers documents for office staff. 4. (computer tech.) Speed-optimized Macintosh software following a simple protocol for tunneling through a TCP/IP internet; network speed is achieved by using turbocharged software; incoming bits spin the turbine that pumps out the outgoing bits.

Overview

TurboGopher is a Macintosh application that we believe is (still!) the fastest Macintosh Gopher client available. Beyond optimizing TurboGopher for raw speed while fetching documents and directories, we turbocharged the user interface by display-

Subsequent Retrievals

Things get easier after the first retrieval. You don't have to use the arcane ftp command to get the Gopher software, you can use Gopher itself.

You will want to familiarize yourself with your gopher client before doing this though. See the chapters that follow to find out about clients.

You can use your Gopher client to connect to the software repository at

```
boombox.micro.umn.edu
```

And choose the files, just as you would with ftp. However, now you can just select the software you want from the menu instead of typing commands to retrieve the file.

Each directory contains gopher for a specific platform. You should retrieve the software from the specific directories that are compatible with your hardware and operating system. There are README files inside of each directory that will help you decide which files to transfer.

Once you find a file that interests you.you should use the GET function of your FTP client. This usually means typing “get filename”, or clicking on the appropriate file with your mouse..

Some clients, notably Unix, VMS, and IBM-PC will reqire you to specify a binary “transfer mode.” You can set this mode by typing **`binary`** at the `ftp>` prompt.

Refer to Figure 1, “Using FTP with Unix,” for an example ftp session to `boombox.micro.umn.edu`

FIGURE 1. Using FTP with Unix

```
% ftp boombox.micro.umn.edu
Connected to boombox.micro.umn.edu
220 boombox FTP server (Version 4.1 Tue Apr 10 05:15:32 PDT 1990)
ready.
Name (boombox.micro.umn.edu:lindner): anonymous
331 Guest login ok, send ident as password.
Password:™
230 Guest login ok, access restrictions apply.
ftp> cd /pub/gopher/Unix
250 CWD command successful.
ftp> binary
200 Type set to I.
ftp> get gopher1.12.tar.Z
200 PORT command successful.
150 Opening BINARY mode data connection for gopher1.12.tar.Z (306512 bytes
226 Transfer complete.
306512 bytes received in 2.4 seconds (1.3e+02 Kbytes/s)
ftp>
```

FTP clients come in many varieties. On a Unix or VMS system the command is called `ftp`. On the Macintosh there are two ftp applications, `Fetch`, and `Xferit`. NCSA has developed a version of ftp for the IBM-PC called 'ftp'. Refer to your local network administrator if you don't have an ftp client.

Once you connect to the boombox server you will want to change to the directory

```
/pub/gopher
```

The different portions for gopher are in this directory. The file 00README contains descriptions of each directory. Refer to Table 1, "Available Gopher Software," on page 10 for a breakdown of the contents of each directory.

TABLE 1. Available Gopher Software

Directory	Description
Mac_server	Gopher server for the Macintosh
Macintosh-TurboGopher	A graphical Gopher client for the Macintosh
NeXT	A graphical Gopher client for NeXTstep
PC_client	A graphical Turbovision based client for PCs running DOS and using the Clarkson/Crynwr packet drivers
PC_server	Two PC Gopher server implementations, one based on Phil Karn's NOS, another using Clarkson/Crynwr packet drivers.
Rice_CMS	A Gopher server and client for VM/CMS systems written at Rice University.
Unix	Gopher for Unix. Includes a Gopher server, full screen client, Xwindow client, and an emacs client.
VMS	Gopher for VMS. Includes a Gopher server and full screen client
VieGopher	A Gopher Server and client for VM/CMS written at Vienna.
gopher_protocol	Descriptions of the Gopher network Protocol are stored here.
incoming	Submitted untested gopher software.
mvs	A Gopher server and client for MVS.
os2	A Gopher client for OS/2.

CHAPTER 2

Getting the Internet Gopher Software

This chapter will show you how to get the Internet Gopher software. You should only need to read the first section *"Getting the Software"* on page 9 and the sections appropriate to your platform.

Getting the Software

You will need to retrieve the Internet Gopher Software from the software repository at the University of Minnesota. Follow these instructions to do so.

The First Retrieval

Most of the software for the Internet Gopher is available on the machine boombox.micro.umn.edu. This machine is on the Internet network. To retrieve the software for the first time you'll need to use an FTP client on a local machine connected to the internet.

FTP, the file transfer protocol, is a quick, reliable and widely popular means of transferring files across the Internet.

If you have questions or comments, please send us electonic mail at gopher@boombox.micro.umn.edu. Or send paper mail to the following address:

Internet Gopher Development Team
100 Union Street SE #152
Shepherd Labs - University of Minnesota
Minneapolis, MN 55455

Or fax us at +1 (612) 625-6817.

- DOS
- Microsoft Windows
- Unix, full-screen, emacs and Xwindow
- VMS full screen.
- NeXTstep
- OS/2
- VM/CMS
- MVS

Gopher Servers

Servers to *publish* information are available for the following platforms:

- Unix
- VMS
- Macintosh
- VM/CMS
- DOS
- MVS

Conventions used in this Manual

This manual uses these conventions:

- The `Courier` font is used to to show sample output of textual clients. **`Bold Courier`** is used whenever you have to enter text or commands.
- In reference to the Unix Server, *gophertop* refers to the directory where the gopher source code is installed.
- Figure boxes are used to show usage.

Support Questions?

We hope you enjoy using Gopher and that it helps you in you to access and publish data quickly and easily.

Using this Manual

You do not need to read this entire manual to use the Internet Gopher. Most likely, you will only need to read the chapters about the Internet Gopher *client* for your software.

If you wish to publish information you will want to pay attention to the chapters referring to the Internet Gopher servers for your system.

The First Steps

See *"Getting the Internet Gopher Software"* on page 9 for information about retrieving and installing the necessary software. Then read the chapter appropriate for your system.

Common Questions

It's a good idea to read through the *"Common Questions"* on page 51. This chapter contains answers to the most frequently asked questions about the Internet Gopher - from installing it to serving data.

As Needed

Chapters x and x may be consulted when you blah blah blah. For an interesting history discussion refer to Chapter X - Growing Gopher.

Platforms Supported

The Internet Gopher works on a number of platforms and operating systems. The operating system for all of the following must have TCP/IP networking support and a connection to a network.

Gopher Clients

Clients to *access* data are available for the following platforms:

- Macintosh

For instance at the University of Minnesota, you might look for a salmon recipe by looking in the *Fun & Games* directory, then in the *Recipes* directory, and then in the *Seafood* directory until you see an item with *salmon* in its title.

Searching

Alternatively, you may want Gopher to do the work for you by using a search item. For instance, at the University of Minnesota you could select a search item called *Search Recipes*. A message prompts you to type in the words you're looking for; you type *salmon*. The server searches the text of a collection of items and lists the ones that have the word salmon in them. You can then examine these items until you find one that contains a recipe that strikes your fancy.

Additionally, some search items will let you specify a complex search expression for your query. If you only had sugar, eggs and chocolate chips, you could search for items that contained all of these words by specifying *sugar and eggs and "chocolate chip"* to the *Search Recipes* item.

Some Example Information

We think the easiest way to get a grasp of the information inside of GopherSpace is to try it. However, for your listing pleasure we have a lisitng of some of the more popular information:

- Weather Forecasts and current conditions for the United States and Canada.
- Recipes
- Movie Reviews.
- Computer Questions and Answers.
- Weather Maps.
- Movies of Chemical reactions.
- E-mail addresses and phone books for most major institutions
- Newspapers and USENET news.
- Full electronic versions of many of the major classics, Shakespeare, Moby Dick, etc.
- Library Catalogs from around the world.
- And many many more!

Telnet Sessions

This item type is a reference to a public telnet connection. Selecting one of these items will connect you to a system using terminal emulation.

Phone Books

This document type is a special version of the search item based upon the Ph/Qi server developed at the University of Illinois.

The phone book search allows you to search on certain fields of a database, such as name, phone number, or address.

Multimedia

Gopher supports a number of multimedia file formats including images, audio and video file formats. Images such as weather maps are available. Audio data, including the presidential debates is available. Movies in Quicktime and MPEG are available.

Formatted Text

Some Gopher servers will allow you to view documents in formats other than text. Postscript is one of the most popular formats for this “Rich Text.”

Finding Information

When using Gopher, looking for information located in other continents is as easy as looking for information residing on a computer in the next room. When using Gopher, you begin at the first or root level of your home Gopher Server. From there you can choose between two methods or locating information: browsing and searching.

Browsing

You may want to follow the paths in Gopher from level to level until you find the data you’re looking for. From the first level, you can choose a topic, which leads to another level, and another, until you finally come to an item that looks interesting.

Gopher combines features of electronic bulletin board services and databases, allowing you to either browse a hierarchy of documents, or search for documents that contain certain words or phrases.

The Internet Gopher software was conceived at the Computer and Information Services department of the University of Minnesota. Software developed at the University of Minnesota is freely distributable for non-commercial puposes.

Types of Information

Gopher supports a diverse range of data, all of which can be accessed by a simple keystroke or click of the mouse. Here are some of the most popular data types:

Directories

The most basic information type in gopher is a directory. A directory is a list of documents. Directories allow easy browsing of information. Items can be organized into specific areas, making it easy to find the information you need.

In addition, a special type of directory called a “link” allows gopher to reference directories on a different computer. This allows gopher to traverse a hierarchy of information residing on multiple machines. These links are transparent, you won’t notice that you’re connecting to another machine.

Text Files

Most of the information in gopher is stored in ASCII text files. These files can be used with most popular computer applications

Search Items

This is a special kind of directory. You can specify any number of keywords to a search item. Only those documents that match the given criteria will show up in the resulting list.

CHAPTER 1

Introducing Gopher

This chapter describes how you can use the Internet Gopher system to easily publish and retrieve information on a network.

Introduction

The internet is a wide vast place with many resources available. Searching, finding and retrieving these resources has been difficult in the past. The Internet Gopher was developed to let an average user access these resources quickly and simply.

What is the Internet Gopher?

All over the world, data is stored on computers, many of which are connected by the Internet, a confederation of computer networks. With Internet Gopher you can easily access publicly available information stored on many of these connected computers.

NeXT Indexing **37**
WAIS Indexing **37**
Preparing to Compile the Distribution **38**
Other Optional Features **40**
Special Modifications for Compatibility **41**
Compiling and Installing the Distribution **42**
Testing **43**

CHAPTER 6 *Go4gw - The Gopher Gateway* ***45***

Overview **45**
Requirements **46**
Installation **46**
Writing New go4gw Gateways **47**

CHAPTER 7 *Common Questions* ***51***

Overview **51**

CHAPTER 3 *TurboGopher* ***13***

Overview **13**
Requirements **14**
Getting Help **14**
Navigating Gopherspace **15**
Alternate Views **17**
Connecting to Terminal-based Services **17**
Advanced Navigation Features **18**
Authenticated Servers **19**
Bookmarks and Bookmark files **20**
Item Attributes **20**
Canceling Slow Network Operations **21**
Starting and Configuring TurboGopher **21**
TurboGopher Options **22**
Miscellaneous **23**
For Experts **23**

CHAPTER 4 *The UNIX Gopher Client* ***27***

Overview **27**
Requirements **27**
Starting the UNIX Gopher Client **28**
Navigating Gopherspace **28**
Special Items **30**
Advanced Navigation Features **31**
Bookmarks **32**
Saving and Downloading **33**

CHAPTER 5 *Installing the Unix Gopher Distribution* ***35***

Overview **35**
Requirements **35**
Compatibility **36**

Contents

CHAPTER 1 *Introducing Gopher* ***1***

Introduction **1**
What is the Internet Gopher? **1**
Types of Information **2**
Finding Information **3**
Some Example Information **4**
Using this Manual **5**
Platforms Supported **5**
Conventions used in this Manual **6**
Support Questions? **6**

CHAPTER 2 *Getting the Internet Gopher Software* ***9***

Getting the Software **9**
Subsequent Retrievals **12**

DISCLAIMER OF WARRANTY

COPYRIGHT NOTICE

Copyright © 1993 by the Regents of the University of Minnesota.

All rights reserved.

Permission to use, copy, modify, and distribute the Gopher software for non-commercial purposes and without fee is hereby granted, provided that the University of Minnesota copyright notices and this permission notice appear in all copies, and that the name University of Minnesota not be used in advertising or publicity pertaining to this software and documentation without specific, written prior permission. The University of Minnesota makes no representations about the suitability of this software and documentation for any purpose. It is provided 'as is' without express or implied warranty.

Commercial use of Gopher requires specific permission from the University of Minnesota; contact the internet gopher development team at <gopher@boombox.micro.umn.edu> for further information.

Printed in the U.S.A.

TRADEMARKS

Internet Gopher is a trademark of the University of Minnesota.
UNIX is a registered trademark of UNIX system Laboratories, Inc.

All other products or services mentioned in this manual are covered by the trademarks, service marks, or product names as designated by the companies who market those products.

ISNB 0-000-000000-0

ABCDEFGHIJ-DO-89

Internet Gopher User's Guide

Editor

Paul Lindner

UNIVERSITY OF MINNESOTA