VieGOPHER

—

A Gopher System for VM/CMS

Gerhard Gonter

June 16, 1993

Abstract

The Internet Gopher Protocol provides a simple, yet very effective method for distributed document search and retrieval. This document describes the usage of the client part of a system for IBM's operating system VM/CMS, called the VieGOPHER.

The system can be distributed freely and is dedicated to the community of cyberspace.

Contents

List of Tables

List of Figures

1 Using the Client

The gopher client is invoked by executing GOPHER. If everything is set up properly, the root menu of your default server should appear within a few seconds. Figure 1 shows a root menu. The beginning of a menu line contains an item marker, <F> indicates a text file and <D> indicates another menu or directory. Table 2 lists all item types that are supported by this client. Please note that other servers might also provide different item types, these are flagged by a marker of the form >*x*<. A list of other know item types is provided in table 2.

1.1 Programmable Function Keys

Basically, the user interface is 'pure' XEDIT with a customized PF-key setting. How the keys are set up is shown in table 1.

PF24 triggers execution of general editor commands just like the ENTER key would do during a normal XEDIT session while the ENTER key has the same function as PF11, that is displaying the selected item. Activating the `Display` function will make the Gopher client fetch the item described by the line of the menu where the cursor is currently in, the result is then displayed as a new XEDIT file. The file containing the menu is then still active, each fetched item adds another file to a so-called XEDIT file ring.

Leaving a file with the `GoBack` function (`PF3`) causes that file to be removed from the editor ring and XEDIT will select another file from the ring for display.

This kind of behaviour might lead the user to a tree-like view of the data space. In reality — whatever that term means — is the user confronted with a quite chaotic, only partially tree like data space. One of the main features of the gopher protocol is the ability to point to other servers. Thus most servers point to the central node in the world-wide gopher network, `boombox.micro.umn.edu`, this server also maintains a list of other known gopher servers. The result is a circle of servers pointing to each other, back and forth.

Figure 1: Root Index of a Gopher Server

```
 GOPHTMP  TMP0     A1  V 122  Trunc=122 Size=20 Line=1 Col=1 Alt=0
====>
<D> WU-Wien Gopher Information Service: Main Menu
gopher.wu-wien.ac.at:70 <1>
<F> About Internet Gopher (gopher.wu-wien.ac.at:70)
<D> Local Info
<F> -----------------------------------------------------------------------
<D> Gophers around the house (Univ. Economics, Vienna)
<D> Gophers around the place (Austrian Universities)
<D> Gophers around the world (and other information services)
<F> -----------------------------------------------------------------------
<D> Early Music List: archives of EARLYM-L@AEARN.bitnet + rec.music.early
<D> CERRO: the Central European Regional Research Organization
<D> WAFE: Widget Athena Frontend
<D> documents
<D> services
<D> external services
<D> pointers                        (things you might also find elsewhere)
<D> life, the NET and everything
<D> The Guide!
<F> -----------------------------------------------------------------------
<D> VieGOPHER: VM/CMS Gopher Version
 1= Help      2=(altpfs)  3= GoBack  4= Disp->BMK   5= Curs->BMK   6= BMK
 7= PageUP    8= PageDN   9= Save   10=            11= Display    12= Circle
```

Table 1: Programmable Function Keys

Key	Label	Description
PF1	`Help`	activate help menu (just for XEDIT)
PF2	`(altpfs)`	assing PF 1-12 with functions of PF 13-24
PF3	`GoBack`	quit the currently displayed item
PF4	`Disp->BMK`	save a bookmark for the currently displayed item
PF5	`Curs->BMK`	save a bookmark for the menu item that is in the line where the cursor currently is
PF6	`BMK`	activate (display) bookmark file as a menu
PF7	`PageUP`	go one page back (up)
PF8	`PageDN`	go one page forward (down)
PF9	`Save`	save the item that is currently displayed
PF10		*unassigned yet, reserved ****
PF11	`Display`	display item of line where cursor is currently
PF12	`Circle`	switch to next active item
PF13	`First`	goto first screen if still active
PF14	`(basepfs)`	assign PF 1-12 with base functions
PF15	`Exit`	leave Gopher immediately
PF16		*unassigned yet, reserved ****
PF17	`Split`	split the screen
PF18	`Join`	join screens
PF19	`PageUP`	go one page back (up)
PF20	`PageDN`	go one page forward (down)
PF21	`ShowRing`	show active items in XEDIT ring
PF22		*unassigned yet, reserved ****
PF23	`Setup`	Setup some parameters for using Gopher
PF24	`Execute`	execute command
ENTER		display item of line where cursor currently is

1.2 File Ring

The XEDIT file ring and the insertion and deletion strategy work like Ariadne's Thread. However, the `Circle` function allows the user to activate the next file in the editor file ring without quitting the current file. The function `ShowRing` (PF20) gives you a list of items in the XEDIT ring, you can select any item by placing the cursor in the appropriate line and pressing ENTER or PF11. All in all, the XEDIT ring system is a perfect way to loose orientation in the Gopher-maze... What was it that I wanted to explain now? Ah yes, bookmarks!

1.3 Bookmarks (BMK)

After exploring Gopher Space for a while you will realize, that an enormous quantity of information is out there. But where exactly did you see that particular piece of invaluable information? You could write down it's 'coordinates'. By the way, coordinates in Gopher Space are three components that are found in a normal Gopher directory or menu: server name, port number and selector string. The item type is useful for your client to know how a particular item should be presented, the fifth component of a directory is only of interest for the user, it is the so-called 'display string' or name and should give you a clue about what you can expect from a particular item.

VieGOPHER, like most other Gopher clients, allows you to take notes about the coordinates of an item you found. This is usually called a bookmark, because it leads you back to the place where you found the information and is not a reproduction of the information itself. You can either take a bookmark for the item that is currently displayed (PF4) or you could request that the item which is in the menu line of the cursor should be transfered to the bookmark list (PF5). The list of your bookmarks is stored in a file named `GOPHER.BOOKMARK` and can be activated by pressing PF6 (the BMK function). The bookmark file is then read and the file is added to the XEDIT file ring.
Notes:

- Save the bookmark file after modifying (editing) it using the FILE command.
- Be careful about activating and editing the bookmark file and taking more bookmarks while the bookmark file is still active in the XEDIT file ring.
- Save bookmarks for WAIS search items from the menu.

1.4 Saving items to a file

Text items can be stored in local files for later processing or printing. This is done by pressing PF9 (save) when displaying the text item of interest. The text will be appended to the file `GOPHER.SAVEFILE`, together with a short header listing the gopher coordinates of the item. Of course, the XEDIT commands `SAVE`, `FILE`, and `PUT` can be entered on the command line as normal.

1.5 Telnet

VieGOPHER supports the telnet types 8 for 'normal' telnet and T for Telnet 3270. IBM's `TELNET.MODULE` is able to handle both types, so the client will call this program by default.

It may be desireable, however, to configure another program to handle type 8 or type T sessions. This can be done either at configuration time by specifying appropriate values for the ROSE-variables `CL_TELNET_8` and `CL_TELNET_T`.

The user can overide this selections by assigning the programs he desires to the global variables `TELNET-8` and `TELNET-T`, both are in the group `GOPHER`.

Please note, the ROSE-variables and the global variables must specify the complete name of the desired program. For example, a user would use VieGOPHER's SETUP function or enter the following command to setup Arty Ecock's VT100 telnet emulator for 'normal' telnet sessions:

```
GLOBALV SELECT GOPHER SETLP TELNET-8 TNVT100 EXEC
```

2 Supported Item Types

Table 2 list all currently supported item types. Please be aware, some servers know item types that this client will not be able to process, other item types (like `w`) are not yet standardized and might be interpreted differently by the server on one side and the client on the other side.

3 Global Variables

The client uses two different types of global variables, permanent ones, for storing setup parameters, in the file `LASTING.GLOBALV` and temporary ones in memory.

3.1 User Setup

Table 3 on page 8 lists all the global variables that are used by the client. They are in the group GOPHER in the file `LASTING.GLOBALV` and can be modified with the SETUP function built into the client (PF23: `Setup`) or by calling the program GOPHSTP.

Table 2: Supported Item Types

type	marker	short	actions on activation
0	`<F>`	File	∘ retrieve ∘ display as plain text
1	`<D>`	Directory	∘ retrieve ∘ display as directory
2	`CSO`	CSO Phonebook[a]	∘ retrieve ∘ display as plain text
7	`<S>`	search	∘ prompt for search string ∘ submit search selector to server[b] ∘ display result as directory
w	`<S>`	search	∘ prompt for search string ∘ submit search selector to server ∘ display result as file
8	`<T>`	Telnet	∘ display selector string as a clue for logging in ∘ start telnet session
T	`<T>`	Telnet 3270[c]	∘ display selector string as a clue for logging in ∘ start telnet session
i	`***`	info	(no action on activation)

[a] under construction, limited support is provided

[b] compatible with WAIS gateway

[c] Type=`T` is now officially declared as 3270 telnet mode. By default, types 8 and T are handled the same way as normal telnet, but this can be configured either at installation time or as a user setup option.

Table 3: Permanent Global Variables for the Client

`HOST`	name of the GOPHER server to be accessed at startup [Default: `gopher.wu-wien.ac.at`]
`PORT`	port of the GOPHER server [Default: `70`]
`ITYPE`	item type of the first item to be fetched [Default: `1` (*directory*)]
`SEL`	selector string of the first item to be fetched [Default: *text*[a]]
`DNAME`	name of the startup menu [Default: *none*]
`BOOKMARK`	name of the bookmark file [Default: `GOPHER BOOKMARK A`]
`SAVEFILE`	name of the file where items may be saved [Default: `GOPHER SAVEFILE A`]
`INFOLEVEL`	level of verbosity, amount of function key description [Default: `5`]
`TELNET-8`	name of the telnet program for items of `Type=8` [Default: `TELNET MODULE`]
`TELNET-T`	name of the telnet program for items of `Type=T` [Default: `TELNET MODULE`]

[a] set during installation as a ROSE definition

Table 4: Temporary Global Variables for the Client

`TMPCNT`	number of the last item fetched (0..??)
`GHOST`	host, port, type and selector of the last item fetched
`GI1:`<*ty*> `GI2:`<*ty*>	directory entry of the item held in file `GOPHTMP.`<*ty*> <*ty*> is the file type of the XEDIT file in the ring
`L`<*ty*>	screen information for active item in XEDIT ring

3.2 temporary variables

Table 4 on page 9 lists the client's temporary global variables, the CMS program GLOBALV keeps them in main memory. These variables can not be accessed directly.

4 Author's Address

I can be reached — for questions, remarks, job offers, etc. — in the following ways:

email:	<Gerhard.Gonter@wu-wien.ac.at> <GONTER@AWIWUW11.BITNET>
snail mail:	Gerhard Gonter University of Economics, Vienna Augasse 2-6 A-1090 Wien Austria, Europe
fax:	+43/1/31336/702
phone:	+43/1/31336/4111