GoFetch!

COLLABORATORS			
	TITLE : GoFetch!		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		July 8, 2022	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

GoFetch!

1.1 GoFetch!.guide

```
                            ,ad8888ba,
              d8"'        '"8b
              d8'
              88                 ,adPPYba,
              88      88888  a8"       "8a
              Y8,        88  8b        d8
               Y8a.   .a88  "8a,   ,a8"
                '"Y88888P"    '"YbbdP"'


   88888888888                          88        88
   88                     ,d            88        88
   88                     88            88        88
   88aaaaa   ,adPPYba,  MM88MMM  ,adPPYba,  88,dPPYba,  88
   88"""""  a8P_____88   88    a8"     ""  88P'   "8a  88
   88       8PP"""""""   88    8b          88      88  ""
   88       "8b,  ,aa    88,   "8a,   ,aa  88      88  aa
   88       '"Ybbd8"'    "Y888 '"Ybbd8"'   88      88  88
```

```
                   Go Fetch! 1.0

                (c) 1999 Ian Chapman

             Written for Zeus Developments

             An FTP/HTTP Client designed to
         download a queue of files in the background.

   This version of Go Fetch! is shareware, all features are enabled except HTTP
        which requires a KeyFile. Please see the section on registering.



              Introduction

              Features
```

1.2 Introduction to Go Fetch!

Introduction

Go Fetch! is a specialised FTP and HTTP client, designed to take the pain out of
downloading lots of files, or very large files. By supplying a simple queue of
files, Go Fetch! will happily just sit in the background and download them for
you, whilst you get on with other tasks like browsing the web.

Scenario 1:

There's a large file on an FTP server you really want, but it's 25MB in size.
You aren't really keen on leaving your machine connected to download a file that
large, because of your phone bill. Well, simply add it to the queue of files in
Go Fetch! and whenever you go online for whatever reason, Go Fetch! will download

some more of the file each time, until you've got the file.

Scenario 2:

There's lots of files on an FTP/HTTP site you want, but they are spread across
different directories. You don't want to have to wait until each file has
finished downloading before getting the next. Well, simply add them to Go Fetch!
and they will all be downloaded while you get on with other tasks.

1.3   Features of Go Fetch!

Features

– Uses MUI GUI.
– No additional MCCs needed.
– Extensive AREXX Support.
– Supports FTP and HTTP (Registered Users Only)
– File Resuming. (FTP only in Version 1.0)
– File Comments.
– Site Directory.
– Profile History.
– CPS Counter
– Integration with your favourite web programs.
– Bubble Help.
– Online Help.
– Context Sensitive Help.
– RFC 959 and basic 1945 Compliancy.
– Uses Fire & Forget technology.
– Log File
– Iconification
– Mini Log which can open on any public screen.

1.4   Requirements

Requirements

– MUI 3.7 or higher
– AmigaOS 3.0 or higher
– BSD Socket compatible TCP/IP Stack (E.g. Miami, Genesis, Termite etc)
– Minimum of 512K RAM Free.
– reqtools.library V38 or higher (Included in distribution)

Testing

Go Fetch! has only been tested with Miami, Miami Deluxe and Termite TCP/IP stacks, ←
    however
it should work with any BSD Socket compatible TCP/IP stack. The following stacks
are BSD Socket compatible.

AmiTCP
AS255r2
Genesis

```
Miami
Miami Deluxe
Termite TCP

Machines Used For Testing

A4000T
Amiga OS 3.1 and OS 3.5
74MB RAM
Cyberstorm Mk 1 060@50Mhz
Cybervision 64/3D running Cybergraphics
Cybervision Scandoubler
Ariadne II Ethernet Card
GVP HC+8 SCSI Card
Buddha IDE Card

A4000T
Amiga OS 3.1 and OS 3.5
154MB RAM
Cyberstorm PPC 604e@233Mhz + 060@50Mhz
Cybervision PPC running Cybergraphics
Ariadne II Ethernet Card
GVP HC+8 SCSI Card

A4000 Desktop
Amiga OS 3.0
16MB RAM
Commodore A3630 030@25Mhz
Cybervision 64/3D running Piccaso 96

A1200
Amiga OS 3.0
6MB RAM
Apollo 1220 020@28Mhz

A2000
Amiga OS 3.1 and 3.5
4 MB RAM
Commodore 2020 (020@14)+MMU+68881
```

1.5 How to install

```
                         Installation
```

There are two installation scripts available, please make sure you use the correct ←
    one.

Install_GoFetch!3x - For Amiga OS 3.0 or greater (Including OS 3.5).

Install_GoFetch!35 - For Amiga OS 3.5 Only

1.6 Menus

Menus

```
Project Menu
************
```

About... (ShortCut: Amiga ?)

    Returns some basic information about Go Fetch!, including what it does, its
    current version, and the author.

About MUI...

    Returns some basic information about your MUI system

MUI Settings

    Allows you to configure your MUI settings, specifically for Go Fetch!

Keyfile Info

    Displays information contained within your keyfile.

Iconify (Shortcut: Amiga I)

    Iconifies Go Fetch! (hides the GUI) and an App Icon will appear on your
    workbench screen. To open the GUI again, double click on the App Icon.

Quit (Shortcut: Amiga Q)

    Quits Go Fetch! and frees any resources.

1.7 Go Fetch! - Main Window

Main Window

General

1. Pressing TAB will cycle through all the gadgets. Pressing return will
activate the currently highlighted gadget.

2. Holding the mouse over a gadget for a short period of time will display a
helo bubble containing a brief description of the gadgets function.

3. Press help in Go Fetch! to display information about the currently selected
window.

MAIN PAGE

Top List View - Profile List

    This contains a list of all the profiles you have added to Go Fetch!. Only

the basic information is shown here, including the protocol the profile uses.

Add Button (Shortcut: A)

Pressing this button will display the add window, allowing you to add a new profile to Go Fetch!. The new profile will be added at the bottom of the list.

Edit Button (Shortcut: E)

Allows you to edit the information for the currently selected profile in the profile listview.

Delete Button (Shortcut: D)

Pressing this button will delete the currently seleced profile. No confirmation action is displayed, so the effect is immediate.

Stop Button (Shortcut: S)

This button is normally only available in the connected state. Pressing it will abort any transfers in operation and stop Go Fetch! from moving onto the next profile.

Go Button (Shortcut: G)

This button will tell Go Fetch! to start running through all your profiles and begin downloading the files. Once Go Fetch! has successfully downloaded a file in a profile, that profile is automatically deleted and Go Fetch! moves onto the next.

Quit Button (Shortcut: Q)

This button simply quits Go Fetch!

Bottom List View - Log

This displays any commands being sent to an FTP/HTTP server, any messages received from an FTP/HTTP server or any information from Go Fetch! itself. Information displayed from Go Fetch! itself, is in bold and preceeded with the hash (#) symbol. Any text sent to the log using an AREXX script is in bold and preceeded with "#RX:"

Save Log Button (Shortcut: V)

Saves all the text in Log Listview as a log file with the name gofetch.log. A filerequester opens, asking you where you would like to save the file.

MiniLog Popup Button

Shows or hides the Minilog. The Minilog provides brief information on what Go Fetch! is currently doing, which allows the user to see at a glance what is going on. The Minilog can be opened on any public screen which to allow monitoring of Go Fetch! from another screen.

Clear Log Button (Shortcut: C)

Clears all the text in the Log Listview.

PROFILE HISTORY PAGE

Profile History Listview

All profiles deleted by Go Fetch! during a file transfer session are displayed
here, including the profiles status.

Clear History

Clears all profiles from the history listview.

PREFERENCES PAGE

This now replaces the tooltypes used in earlier versions of Go Fetch! Any ←
    tooltypes
set will be ignored.

Anonymous Password

This is default password used for connecting to FTP servers anonymously. ←
        Convention
dictates that this should usually be your E-Mail address. If no preferences
are found, then the default is nobody@nobody.com.

Default Download Directory

The default location to download files, for both FTP and HTTP transfers. If
no preferences are found, then the default is RAM:

MiniLog Screen

The public screen name of a screen on which the Go Fetch! Minilog Window
should open. If the screen is closed when the Minilog Window is opened, it
will be displayed on the systems default public screen. (Usually Workbench).
If no preference file is found, then the default screen is Workbench.

Beep On Completed File Transfer

When selected Go Fetch! will beep, or play the system sample upon completion
of a file transfer.

Save Prefs

Saves the current preferences.

1.8 Go Fetch! - Add Window

                            Add Window

General

1. Pressing TAB will cycle through all the gadgets. Pressing return will
activate the currently highlighted gadget.

2. Holding the mouse over a gadget for a short period of time will display a
helo bubble containing a brief description of the gadgets function.

3. Press help in Go Fetch! to display information about the currently selected
window.

FTP PAGE

Use this tab, to add a new FTP profile.

Site

    The URL of the site you want to enter. This can either be a standard text
    URL such as ftp.whatever.com or the site IP address. Do not include any path
    information or filename here.

Add Site Button (Right Arrow)

    Pressing this button will add the site currently in the site text box,
    into the site directory. Please note, permanent changes to the site
    directory will only be made, once the add window has been closed.

Popup Site Directory Button

    Pressing this will display the site directory. If the site directory is
    currently displayed, pressing it again will hide the site directory. The
    site directory is intended as a covenient 'database' to add URLs which
    you commonly use to prevent you from typing them in.

Site Listview (If Displayed)

    This displays all the URLs, currently in the site directory. Double clicking
    on any URL in the site directory will put that URL into the Site Text Box.

Delete Site from Directory Button (If Displayed, ShortCut D:)

    Pressing this button, will delete the currently selected URL from the site
    directory. Please note, permanent changes to the sited irectory will only be
    made, once the add window has closed.

Port Textbox

    The FTP port of the site you want to access. This is almost always 21 so you
    should have rare occasion to change it. If you don't know what this is for

then you probably won't ever need to change it.

Username Textbox

The login name you need to use to log into the site, or anonymous if Anon
Login is ticked.

Anon Login CheckMark

For many FTP sites, you don't need a specialised account to access their
files because they allow anonymous login. If you don't have an account on
the FTP site, tick this box to perform an anonymous login. Username will
contain anonymous and password will contain the default anonymous password
you specified as a Tool Type. This is usually your E-Mail address.

Password TextBox

This is the password you need to login into the site. If Anon Login is
ticked, it will contain your default anonymous password.

Remote Path Textbox

This should contain the path/location of the file on the remote FTP server.
Remember, most FTP servers are UNIX systems, so the path should be case
correct too. For example, /pub/system is not the same as /PUB/system .

Filename Textbox

This contains the complete filename of the file you wish to download.
Remember, most FTP servers are UNIX systems, so the filename should be case
correct. For example, gofetch.lha is not the same as GoFetch.lha .

Download Textbox

This contains the path where you want to download the file to.

File Icon

Pressing this will bring up a file requestor asking where you want to
download files to. Once a directory has been chosen, the path will
automatically be entered into the Download Textbox.

OK Button

Pressing this will add the new FTP profile to Go Fetch!. It will appear at the
bottom of the profile listview.

Cancel Button

Pressing this will discard the current profile and close the Add Window.

HTTP PAGE (Active only for registered users)

Use this tab to add a new HTTP profile.

Site

    Complete URL including the path and filename. For example,
    www.whatever.com/files/myfile.lha. DO NOT prefix it with HTTP://

Port

    The HTTP port of the site you want to access. The default HTTP port is 80,
    and you should have rare occasion to change it. If you don't know what this
    is for, then you probably won't need to change it.

Download

    The location in which to save the file.

File Icon

    Pressing this will open a file requestor and allow you to choose a location.
    The location is automatically entered into the Download textbox.

OK

    Pressing OK will add the current HTTP profile to the profile listview and
    close the add window. The profile is automatically saved.

Cancel

    Pressing cancel will discard the HTTP profile and close the add window.

1.9 Go Fetch! - Edit Window

                                    Edit Window

General

1. Pressing TAB will cycle through all the gadgets. Pressing return will
activate the currently highlighted gadget.

2. Holding the mouse over a gadget for a short period of time will display a
helo bubble containing a brief description of the gadgets function.

3. Press help in Go Fetch! to display information about the currently selected
window.

FTP PAGE

If the current profile you are editing is an FTP profile, then all the gadgets
in the HTTP TAB will be ghosted.

Site Textbox

   The URL of the site you want to enter. This can either be a standard text
   URL such as ftp.whatever.com or the site IP address.

Add Site Button (Right Arrow)

   Pressing this button will add the site currently in the site text box,
   into the site directory. Please note, permanent changes to the site
   directory will only be made, once the edit window has been closed.

Popup Site Directory Button

   Pressing this will display the site directory. If the site directory is
   currently displayed, pressing it again will hide the site directory. The
   site directory is intended as a covenient 'database' to add URLs which
   you commonly use to prevent you from typing them in.

Site Listview (If Displayed)

   This displays all the URLs, currently in the site directory. Double clicking
   on any URL in the site directory will put that URL into the Site Text Box.

Delete Site from Directory Button (If Displayed, ShortCut D)

   Pressing this button, will delete the currently selected URL from the site
   directory. Please note, permanent changes to the sited irectory will only be
   made, once the edit window has closed.

Port Textbox

   The FTP port of the site you want to access. This is almost always 21 so you
   should have rare occasion to change it. If you don't know what this is for
   then you probably won't ever need to change it.

Username Textbox

   The login name you need to use to log into the site, or anonymous if Anon
   Login is ticked.

Anon Login CheckMark

   For many FTP sites, you don't need a specialised account to access their
   files because they allow anonymous login. If you don't have an account on
   the FTP site, tick this box to perform an anonymous login. Username will
   contain anonymous and password will contain the default anonymous password
   you specified as a Tool Type. This is usually your E-Mail address.

Password TextBox

   This is the password you need to login into the site. If Anon Login is
   ticked, it will contain your default anonymous password.

Remote Path Textbox

   This should contain the path/location of the file on the remote FTP server.
   Remember, most FTP servers are UNIX systems, so the path should be case

correct too. For example, /pub/system is not the same as /PUB/system .

Filename Textbox

This contains the complete filename of the file you wish to download.
Remember, most FTP servers are UNIX systems, so the filename should be case
correct. For example, gofetch.lha is not the same as GoFetch.lha .

Download Textbox

This contains the path where you want to download the file to.

File Icon

Pressing this will bring up a file requestor asking where you want to
download files to. Once a directory has been chosen, the path will
automatically be entered into the Download Textbox.

OK Button

Pressing this will make the necessary changes to the selected profile.

Cancel Button

Pressing this will discard any changes and close the edit window.

HTTP PAGE (Active Only for Registered Users)

If the current profile you are editing is an HTTP profile, then all gadgets in
the FTP TAB will be ghosted.

Site

Complete URL including the path and filename. For example,
www.whatever.com/files/myfile.lha. DO NOT prefix it with HTTP://

Port

The HTTP port of the site you want to access. The default HTTP port is 80,
and you should have rare occasion to change it. If you don't know what this
is for, then you probably won't need to change it.

Download

The location in which to save the file.

File Icon

Pressing this will open a file requestor and allow you to choose a location.
The location is automatically entered into the Download textbox.

OK

Pressing OK will save the changes to the HTTP profile and close the edit

window

Cancel

    Pressing cancel will discard any changes and close the edit window.

1.10   Go Fetch! - Transfer Window

Transfer Window

Transfer Gauge

    This displays how much of the file you have received so far in Kilobytes, as
    well as the percentage received.

CPS Rate

    This displays your CPS rate, aka the speed of the transfer. Go Fetch! takes
    a sample every five seconds and calculates the average cps rate over those
    five seconds.

Abort Button (Shortcut: A)

    Pressing this will abort the current file transfer, Go Fetch! will close the
    connection, then continue onto the next profile.

Stop Button (On Main Window)

    Pressing this will abort the current connection, Go Fetch! will close the
    connection to the server, and NOT continue with any of the other profiles.

1.11   Go Fetch! - Arexx Commands

Arexx Commands

For examples of how to use AREXX with Go Fetch!, please see included AREXX
scripts.

AREXX PORT: GOFETCH

These returncodes apply to all Arexx Commands
-4 = Command received wrong number of parameters
-3 = Command received parameter of wrong type
-2 = Out of Memory (except where stated)

Command: addanonprofile
Parameters: SITE/A,PORT/A,REMOTEPATH/A,FILENAME/A,DOWNLOADDIR/A
ReturnCodes: 0 = Success
Description:
Adds a new FTP profile to Go Fetch!. The profile will perform an anonymous
login, using your default anonymous password. The profile is automatically

saved.


```
Command:        addhttpprofile
Parameters:     SITE/A,PORT/A,DOWNLOADDIR/A
ReturnCodes:    0 = success
                -5 = Go Fetch! is not registered.
Description:
Adds a new HTTP profile to Go Fetch!. SITE should contain a complete URL
including path and filename, for example, www.whatever.com/files/file.lha. The
profile is automatically saved.
Note: Requires registered version of Go Fetch!
(See: addprofile, edithttpprofile
```


```
Command:        addprofile
Parameters:     SITE/A,PORT/A,USERNAME/A,PASSWORD/A,REMOTEPATH/A,FILENAME/A, ←
    DOWNLOADDIR/A
ReturnCodes:    0 = success
Description:
Adds a new FTP profile to Go Fetch!. The profile will perform a named login,
using the login name and password you specify. The profile is automatically
saved.
(See: editprofile, addhttpprofile)
```


```
Command:        addsite
Parameters:     SITE/A
ReturnCodes:    0 = Success
                -1 = Failed, could not open site directory file.
                -2 = Failed, Add or Edit window was open.
Description:
Adds an FTP URL to the site directory. This should be the domain only and not
include any path details of filename details. You cannot add a site, if the Add
or Edit Window is open.
```


```
Command:        annotate
Parameters:     TEXT/A
ReturnCodes:    0 = Success
Description:
Writes directly to the Go Fetch! log listview. This can be useful in cases where
AREXX scripts need to give any output. Please note that any strings longer than
74 characters will be truncated. All strings sent using annotate will appear in
bold and are prefixed with "#RX:". Special codes can be inserted into the text
to provide special formatting, each code used will count as 1 character. These are ←
    :

(The Escape Character (Ascii 27, $1B) can be displayed in most text editors by
pressing CTRL [)

ESCu - Underline
ESCb - Bold
ESCi - Italics
ESCn - Set back to normal.
ESC<x> - Where X is a colour between 2 and 9.
```

Note: NO OTHER Escape Sequences should be used.

Command: clearhistory
Parameters: NONE
ReturnCodes: 0 = success
Description
Clears the profile history.

Command: clearlog
Parameters: NONE
ReturnCodes: 0 = success
Description:
Clears the Go Fetch! log listview.

Command: deleteprofile
Parameters: PROFILE/N
ReturnCodes: 0 = success
               -1 = Profile doesn't exist.
Description:
Deletes the profile passed by the profile number. Please note that profiles are
are numbered starting at 0. For example if there are 8 profiles in Go Fetch!
then they are numbered from 0 to 7.
Warning: Under no circumstances should you pass a negative profile number.
(See: profiles)

Command: dlstatus
Parameters: NONE
ReturnCodes: 0 = In Unconnected State
               1 = In Connected State (But Not Downloading)
               2 = Downloading FTP File.
               3 = Downloading HTTP File.
Description:
Tells you what state Go Fetch! is currently in.

Command   edithttpprofile
Parameters: PROFILE/N,SITE/A,PORT/A,DOWNLOADDIR/A
ReturnCodes: 0 = success
               -1 = Profile Doesn't Exist
               -5 = Go Fetch! isn't registered.
Description:
Edits the currently chosen profile. Please note that profiles are numbered from
0 onwards, for example if there are 8 profiles in Go Fetch! then they are
numbered 0 to 7. The edited profile is automatically saved, and the profile
listview is updated to reflect the changes.
Note: Requires Registered version of Go Fetch!
Warning: Under no circumstances should you pass a negative profile number.
Warning: DO NOT use this command to edit an FTP profile. It may cause adverse
         effects. Use the command returnprotocol to check first.
(See: profiles, addhttpprofile, editprofile, returnprotocol)

Command: editprofile

```
Parameters:    PROFILE/N,SITE/A,PORT/A,USERNAME/A,PASSWORD/A,REMOTEPATH/A,FILENAME ↩
    /A,DOWNLOADDIR/A
ReturnCodes:   0 = success
               -1 = Profile doesn't exist.
Description:
Edits the currently chosen profile. Please note that profiles are numbered from
0 onwards, for example if there are 8 profiles in Go Fetch! then they are
numbered 0 to 7. The edited profile is automatically saved, and the profile
listview is updated to reflect the changes.
Warning: Under no circumstances should you pass a negative profile number.
Warning: DO NOT use this command to edit an HTTP profile. It may cause adverse
         effects. Use the command returnprotocol to check first.
(See: profiles, addprofile, edithttpprofile, returnprotocol)



Command:       getanon
Parameters:    NONE
ReturnCodes:   0 = success
ReturnString:
Description:
Returns the users default anonymous password, as set in the preferences.



Command:       getdownloadpath
Parameters:    NONE
ReturnCodes:   0 = success
ReturnString:  Default Download Directory
Description:
Returns the users default download directory, as set in the preferences.



Command:       gofetch
Parameters:    NONE
ReturnCodes:   0 = Success
Description:
Simulates the GO button. Go Fetch! will start downloading files and working
through the profiles.



Command:       iconify
Parameters:    NONE
ReturnCodes:   0 = success
Description:
Iconifies Go Fetch!
(See: uniconify, iconifystatus)



Command:       iconifystatus
Parameres:     NONE
ReturnCodes:   0 = uniconified
               1 = iconified
Description:
Returns whether Go Fetch is iconified.
(See: iconify, uniconify)



Command:       lockgui
```

```
Parameters:   NONE
ReturnCodes:  0 = success
Description:
Locks the Go Fetch! main window from receiving user input. This command is nested,
therefore if you tell the window to lock twice, you must also tell it to unlock
twice.
Warning: Use with caution. Misuse could lock the user out of the program
         permanently.
(See: unlockgui)


Command:      openaddwindow
Parameters:   NONE
ReturnCodes:  0 = success
Description:
Opens the profile add window, simulates the user pressing the ADD button.
Warning: This command will return success (0), regardless of whether the
         window opened or not. Be prepared for it.


Command:      openeditwindow
Parameters:   NONE
ReturnCodes:  0 = success
Description:
Opens the edit window, simulates the user pressing the EDIT button. Please note
that the edit window will not open if a profile isn't active.
Warning: This command will return success (0), regardless of whether the window
         opened or not. Be prepared for it.
(See: setactiveprofile, profiles)


Command:      profiles
Parameters:   NONE
ReturnCodes:  Number of Profiles
Description:
Returns the number of profiles in Go Fetch!. For example, if there are 8
profiles in Go Fetch!, then this command will return 8.


Command:      quitgofetch
Parameters:   NONE
ReturnCodes:  0 = success
Description:
Quits Go Fetch!


Command:      registered
Parameters:   NONE
ReturnCodes:  0 = Not Registered.
              1 = Registered
Description:
Returns whether Go Fetch! has been registered or not. This may be required as
AREXX commands relating to HTTP are for registered users only. This may extend
to new commands in the future.


Command:      release
```

```
Parameters:     NONE
ReturnCodes:    Release Number
Description:
Returns the release number of Go Fetch!. Each public release of Go Fetch! will
have it's release number incremented by 1, independant of the program version.


Command:        reloadprofiles
Parameters:     NONE
ReturnCodes:    0 = success
Description:
Forces Go Fetch! to reload the profiles from the profiles file. This has no
real practical use because Go Fetch! automatically updates the profiles
whenever an addition or edit is made. This command may be useful to those who
feel confident enough to edit the profiles file directly, however this is highly
unadvised.


Command:        returnprotocol
Parameters:     PROFILE/N
ReturnCodes:    0 = Profile is an FTP Profile
                1 = Profile is an HTTP Profile
Description:
Returns the protocol a profile is set to use. Please note, profiles are numbered
from 0 onwards. For example, if there are 8 profiles, then they will be numbered
0 to 7.
Warning: Under no circumstances should you pass a negative profile number.
(See: edithttpprofile, editprofile)


Command:        savelog
Parameters:     PATH/A
ReturnCodes:    0 = success
                -1 = Unable to save logfile. Check your path.
Description:
Saves the Go Fetch! log with a filename of gofetch.log at the chosen path.


Command:        setactiveprofile
Parameters:     PROFILE/N
ReturnCodes:    0 = success
                -1 = Profile doesn't exist
Description:
Highlights the profile entry in the profile listview passed by the profile
number. Please note that profiles are numbered starting at 0. For example if
there are 8 profiles in Go Fetch! then they are numbered from 0 to 7.
Warning: Under no circumstances should you pass a negative profile number.
(See: profiles)


Command:        setminilog
Parameters:     STATUS/N
ReturnCodes:    0 = success
                -1 = Failure, window is already open/closed
Description:
If STATUS is 1 (actually =>1) then it will try to open the minilog window.
If STATUS is 0 (actually <=0) then it will try to close the minilog window.
```

```
Command:       stop
Parameters:    NONE
ReturnCodes:   0 = success
Description:
Aborts the current file transfer, and stop Go Fetch! from moving onto the next
profile.


Command:       uninconify
Parameters:    NONE
ReturnCodes:   0 = success
Description:
Uniconifies Go Fetch!
(See: iconify)


Command:       unlockgui
Parameters:    NONE
ReturnCodes:   0 = success
Description:
Unlocks the Go Fetch! main window to allow user input after the lockgui command
has been issued. This command is nested, therefore if you tell the window to
lock twice, you must also tell it to unlock twice.
(See: lockgui)


Command:       windowopen
Parameters:    NONE
ReturnCodes:   0 = No Windows Open
               1 = Add Window Open
               2 = Edit Window Open
Description:
Returns whether any windows are currently open (Add, Edit).



                        Standard MUI Arexx Commands

Command:       HELP
Parameters:    FILE/A
ReturnCodes:   0 = success
Description:
Lists GO Fetch Arexx Commands into the file specified.


Command:       HIDE
Parameters:    NONE
ReturnCodes:   0 = success
Description:
Iconifies Go Fetch!


Command:       INFO
Parameters:    ITEM/A
ReturnCodes:   0 = success
```

```
Returns requested string in result.
Description:
Where ITEM is:
TITLE = Title of Application
AUTHOR = Author of Application
COPYRIGHT = Copyright Message
DESCRIPTION = Description of Go Fetch
VERSION = Current Version
BASE = Name of AREXX Port
SCREEN = Name of public screen


Command:      QUIT
Parameters:   NONE
ReturnCodes:  0 = success
Description:
Quits Go Fetch!


Command:      SHOW
Parameters:   NONE
ReturnCodes: 0 = success
Description:
Uniconfies Go Fetch!
```

1.12 Arexx Scripts

                              Arexx Scripts

These scripts are designed to integrate Go Fetch with other applications, for
example Aweb-II which allows you to automatically create profiles. Scripts for
the other browsers such as Voyager and IBrowse as soon as they have decent AREXX
ports.

If you have any AREXX scripts which integrate Go Fetch! into other software,
please feel free to E-Mail me them and you'll receive a FREE keyfile for future
versions of Go Fetch!

Please note that any submitted AREXX scripts will be distributed with future
versions of Go Fetch!. Upon submitting the author gives their permission for Ian
Chapman and Zeus Developments to distribute and modify the code. Any
modifcations would be commented as so in the source. The original author will
have full credits in the source.

                         The Script(s)

                 browser.rexx
                 by Marc Bradshaw

                 StartGoFetch.miami
                 by Marc/Ian

                 StopGoFetch.miami
                 by Marc/Ian

```
                        AmFTPtoGoFetch.rexx
                        by Mike King

                        YAMtoGoFetch.rexx
                        by Mike Cardwell

                        GoFetch.thor
                        by Neil Bothwick

                        GoFetch.br
                        by Neil Bothwick

                        AminetFetch.rexx
                        by Olivier Fabre

                        AminetExtract.rexx
                        by Olivier Fabre

                        batch2gofetch.rexx
                        by Stephen Bridges

                        yam2gofetch.rexx
                        by Stephen Bridges
```

1.13 browser.rexx

```
                        browser.rexx
```

Notes:
This script has only been tested on Aweb-II version 3.0 and the AWeb version
supplied with AmigaOS 3.5 and therefore I cannot guarantee it's compatibility
with different versions. The installation procedure may differ depending on
your version of Aweb.

There are two ways in which you can use the script within Aweb. For maximum
flexibility you probably want to use both.

Use: Method 1
Installation in Aweb via the AREXX Menu.

Method 1 uses the AREXX script in the AREXX menu of Aweb. This allows you to
mark text (as if you were doing a copy & paste) and selecting the AREXX script
from the menu, it will automatically create a new profile.

1. Open Settings Menu
2. Choose GUI Settings
3. Select the AREXX tab.
4. Press ADD and fill in the following fields.

```
    Title = Send to Go Fetch!
    Macro = <path to browser.rexx> AWEBSEL PATH <path to the GoFetch! Executable>
    n.b. the PATH argument is optional, if GoFetch! is in the command path then
    the script will find it, if not you must specify its path here.
```

Use: Method 2
Installation in Aweb via PopUp Menus

Method 2 uses the AREXX script in a popup menu. This allows you to press the
right mouse button over a link and send the link to Go Fetch! and a profile will
be created for you.

1. Open Settings Menu
2. Choose GUI Settings
4. Select PopUp TAB
3. Press ADD and fill in the following fields
```
        Type = Link Popup Menu
        Title = Send to Go Fetch!
        Command = RUN <path to browser.rexx> URL %u PATH <path to GoFetch!  ←
            Executable>
        n.b. the PATH argument is optional here as above.
```

Use from the command line

rx browser.rexx [AWEBSEL | URL <url>] [TO <directory>] [PATH <path>] [?]

? – Request This Help Text

AWEBSEL – if run from aweb, grabs selected text as url
          should not be used with the URL keyword

URL <url> – the url to add as a profile
            should not be used with the AWEBSEL keyword

TO <directory> – direcroty to download to
                 if not supplied the default as defined in
                 GoFetch! will be used.

PATH <path> – GoFetch! Path The Path to the GoFetch! executable
              if not supplied the script will search the current
              command path for GoFetch!

For Example:

rx browser.rexx URL ftp://site.com/dir/anotherdir/file.lha
rx browser.rexx URL ftp://me@secret@site.com/yetanotherdir/kewl.txt Path SYS: ←
    Internet/GoFetch!/GoFetch!
rx browser.rexx URL ftp://site.com/file.lha TO Ram:

If Go Fetch! is not running when the script is called, it will attempt to start
Go Fetch! for you, by checking in the standard command path. You MUST make sure
that GoFetch! is in the command path or you have specified the path to the
GoFetch! Executable using the PATH option if you want this feature to work.

Return Codes
Browser.rexx returns the following values.

```
0 = Success
1 = Go Fetch! is not running and could not be started
2 = Go Fetch! is not running and couldn't be located.
3 = Go Fetch! is not running and the user chose not to start it.
4 = Unsupported Protocol (Eg trying to use HTTP/FILE/GOPHER etc)
5 = Invalid URL

-ve = Add Profile return value
-4 = Wrong number of parameters passed to Go Fetch!
-3 = Parameter of wrong type passed to Go Fetch!
```

1.14   startgofetch.miami

StartGoFetch.miami & StopGoFetch.miami

These scripts are designed to automatically start and stop GO Fetch! when you go
online. If the AREXX script finds profiles contained within Go Fetch! it will
tell Go Fetch! to GO, otherwise it will Quit Go Fetch!. When you go offline, the
AREXX scripts tells Go Fetch! to Stop and Quit.

Miami Deluxe (DX) Installation
1. Choose Interfaces.
2. Highlight the interface you want to use Go Fetch! with when you go online.
3. Press Edit
4. Press Events
5. Change "Active Offline" to the location of the StopGoFetch.miami script
6. Change the script type to AREXX
7. Repeat this for "Passive Offline"
8. Change "Online" to the location of the StartGoFetch.miami script, followed by
   the location of your Go Fetch! executable.
9. Change the script type to AREXX
10. Press OK and Save your settings.

Miami Installation

1. Choose Events
2. Change "Active Offline" to the location of the StopGoFetch.miami script
3. Change the script type to AREXX
4. Repeat this for "Passive Offline"
5. Change "Online" to the location of the StartGoFetch.miami script, followed by
   the location of your Go Fetch! executanle
6. Change the script type to AREXX
7. Save your settings.

1.15   amftptogofetch.rexx

AmFTPtoGoFetch.rexx
by Mike King

This script reads in a batch file from amFTP and adds each
entry from it into GoFetch! Profiles so that GoFetch!
can be used to download the files.

Usage:

You must have a profile in Amftp for the site you
want to download all the files from.

Logon to that site from amftp and save a batch list
of all the files you want to download.

Name the Batch list the EXACT name as the Profile used
in amftp's default batch directory.

Then Run this script using the batch list filename as the
argument.

    Example:

    Save a batch of files from aminet into a file name AMINET
        in amftp's batches/ directory. Then run the script
    Batch->GoFetch AMINET

Program:

The program Uses the argument to load the file from amftp's batch directory.   ←
    It
always uses the batch directory. Then it starts reading files from
batch list. It finds the profile in amftp that matches the
filename to get all the conenction information from.

Then it creates a directory structure for the files in your download
location that matches the deepest directory name from the site.
This keeps everything organised.

Then it adds all the files as profiles into GoFetch!

NOTE:
    You will need to alter the paths in this file if you want it to
    automatically startup GoFetch! and amftp when needed.

1.16 Tool Types

                                Tool Types

From version 1.0, the Tooltypes are no longer used. Please use the preferences
page within Go Fetch! to choose your settings.

1.17 Bug Reports

                              Bug Reports

Go Fetch! has been extensively tested, however if you discover a bug, please
submit a bug report and include the following information

Machine

Machine Description
(Including Model, Processor, RAM, Cards, etc)

AmigaOS Version

TCP/IP Stack you are using
(Including version)

Any Enforcer/Cyberguard/Segtracker/Mungwall output

A good clear desciption of the bug.

If possible, a log file created from Go Fetch!

Details on the FTP site you were using, and the file you were tring to download.

Send bug reports to ian.chapman@bleach.demon.co.uk

1.18 Program History

                              History

Thanks to all those who have registered and sent it suggestions & Bug Fixes.

Version 0.9b

    First Public Beta Release

Version 0.95b

    Second Public Beta Release
    Bug Fixes:
        -Now Snapshots Transfer Window.
        -Now Snapshots Add Window
        -Now Snapshots Edit Window
        -Anon Checkmark is now set if profile is anonymous
        -Changed timeout to 25 seconds (was 40)
        -Fixed continuous downloading. Previously it would not continue onto the
        next profile if the main window was not selected. This was due to a quirk
        in MUI.
        -Moved location of profiles file to the program directory. This might
        resolve conflicts with HappyENV.
        -Fixed RC 421 Problem. Previously if Go Fetch! received a code 421 on
        login, it would send QUIT then get stuck in a Busy Loop.
        (Jostein Klemmetsrud)
    New Features:
        -Adds comments to downloaded files.

```
        -Added Site Directory
    New Arexx Commands
        addanonprofile
        addsite
        gofetch
        stop
        windowopen
    New ARexx Scripts
        StartGF.miami
        StopGF.miami
        AmFTPtoGoFetch.rexx
        browser.rexx - UPDATED


Version 0.97b

    Internal Release Only

 Version 1.0

    First Shareware Release

    Bug Fixes:
        - Now REALLY snapshots the windows. The problem was due to a pointer
          being passed instead of a unique identifier, so consequently MUI would
          usually forget the window positions.
        - FTP routines are a little more robust, should fix the problem with War
          FTP of logging out prematurely.
    New Features
        - Added basic HTTP support (Registered Users Only)
        - Added a Minilog window which can be opened on any public screen.
          Provides brief information on what Go Fetch! is doing.
        - Added preferences tab. (Replaces Tooltypes)
        - Added profile history tab which records the status of used profiles.
        - Improved GUI look.
        - CPS Counter
    New Arexx Commands
        addhttpprofile
        annotate
        clearhistory
        edithttpprofile
        returnprotocol
        registered
        setminilog
    New Arexx Scripts
        AmFTPtoGoFetch.rexx (Mike King)
        AminetExtract.rexx (Olivier Fabre)
        AminetFetch.rexx (Olivier Fabre)
        batch2gofetch.rexx (Stephen Bridges)
        GoFetch.br (Neil Bothwick)
        GoFetch.thor (Neil Bothwick)
        yam2gofetch,rexx (Stephen Bridges)
        YAMtoGoFetch.rexx (Mike Cardwell)
```

1.19 Future

Future

I would like to add the following features to future versions of Go Fetch! If
you have any additional suggestions, then please feel free to E-Mail me.

- Resume for HTTP sites that support it.
- Multiple Simultaneous Downloading
    (Downloading from several sites at the same time)
- Bandwidth Setting
    (Set a maximum bandwidth Go Fetch! should use)
- Locale Support
- Keep connection alive if the next file is on the same server
- Profile Reorganisation

1.20 Legal

Legal

ZEUS DEVELOPMENTS AND IAN CHAPMAN MAKE NO REPRESENTATIONS OR WARRANTIES IN ANY
FORM WITH RESPECT TO ANY SOFTWARE IN THE "GO FETCH!" PACKAGE AND SPECIFICALLY
DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTIES WITH REGARDS TO THE MERCHANTABILITY,
TITLE OR FITNESS FOR A PARTICULAR PURPOSE, EXCEPT WHERE REQUIRED UNDER THE LAWS
OF THE UNITED KINGDOM OF GREAT BRITAIN AND NORTHERN IRELAND. IAN CHAPMAN
RESERVES THE RIGHT TO MODIFY, REPLACE, EXCLUDE OR DISCONTINUE ANY AND ALL PARTS
OF THE "GO FETCH!" PACKAGE AT ANY TIME, WITHOUT OBLIGATION OR PRIOR NOTIFICATION
TO ANY PERSON(S) OR ENTITY. IAN CHAPMAN OR ZEUS DEVELOPMENTS WILL IN NO WAY BE
LIABLE TO YOU FOR DAMAGES ARISING OUT OF THE USE OR MISUSE OF THE PROGRAM AND
RELATED FILES (INCLUDING, BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED
INACCURATE OR LOSSES SUSTAINED BY ANY PERSON(S) OR ENTITY, OR DAMAGE TO
HARDWARE), EVEN IF ANY PERSON(S) OR ENITY HAVE BEEN ADVISED OF SUCH DAMAGES.

ALL TRADEMARKS ARE RECOGNISED AS BELONGING TO THEIR RESPECTIVE OWNERS.

THE FOLLOWING LICENSE APPLIES TO THE PRODUCT "GO FETCH!" AND RELATED FILES IN
THE "GO FETCH!" PACKAGE, WRITTEN BY IAN CHAPMAN FOR ZEUS DEVELOPMENTS.

* You may only distribute verbatim copies of the original distribution as
released by Ian Chapman and/or Zeus Developments. All executables, documentation,
examples and related files must be intact. The only exception is that the entire
original distribution may be compressed in another form, providing it is fully
intact. Ian Chapman reserves the right to disallow the use of any and all parts
of the distribution by any person(s) or entity,

* You may not modify any binary or documentation contained within the package,
except for purposes of compression. Disassembling, decompiling, re-sourcing or
otherwise reverse engineering the program is strictly prohibited.

* By the use and distribution of any part of the distribution, you agree to the
license in full.

* "Go Fetch!" and any "Go Fetch!" distribution may not be sold, or re-sold
without written permission from Ian Chapman.

* For use of the "Go Fetch!" program and distribution, outside of this license
agreement, permission must be obtained from Ian Chapman.

* Go Fetch! may not be used in any commercial distributions or enterprises
without witten permission. The exceptions to this are the inclusion on any
Aminet CD's or Magazine Coverdisks and CoverCDs.

* The duplication or distribution of keyfiles is strictly prohibited, however
  one duplicate is allowed for backup purposes only. Keyfiles may not be sold,
  resold or transferred to any person(s) or entity without written permission.

* FREE keyfiles remain the property of Ian Chapman and the right to use those
  keyfiles main be withdrawn by Ian Chapman, at any time without reason or
  explanation of any kind, to any person(s) or entity.

1.21 Contacting the Author

                                Contact

E-Mail: ian.chapman@bleach.demon.co.uk
Fidonet: 2:256/607.11

Go Fetch! Web Site
www.bleach.demon.co.uk/gofetch/

Zeus Developments Web Site
www.bleach.demon.co.uk/zeus/

Big Book of Amiga Hardware Webpages
www.amigaworld.freeserve.co.uk

1.22 yamtogofetch.rexx

                            YAMtoGoFetch.rexx
                             by Mike Cardwell

To integrate with YAM.
1.) Go to Configuration in the Settings menu.
2.) Select "ARexx"
3.) Scroll to the bottom, and click on "When double-clicking an url"
4.) Select Arexx from the pull down menu.
5.) Type in the full path of Yam-GoFetch (Including the filename)
6.) Both tick boxes want to be unselected
7.) Fill in requested information below

Now. When double clicking an ftp:// url, the download information will
be sent to GoFetch. It will only work with URL's ending with a filename.

Please see the AREXX script for additional information.

1.23 gofetch.thor

```
                         GoFetch.thor
                       by Neil Bothwick


Designed for use in the E-Mail/Fidonet Package THOR.

Runs GoFetch.br for the current system
Can be called from the Arexx menu or as a system exit event

Please see the AREXX script for additional information.
```

1.24 gofetch.br

```
                         GoFetch.br
                       by Neil Bothwick

   GoFetch.br will transfer files selected for download in THOR
   from the Events editor to the GoFetch! profiles list.

   Usage: GoFetch.br SystemName/A

   If you want to do the transfer from within THOR, run GoFetch.thor
   from the Arexx menu or as the system Exit command.
   GoFetch.thor simply determines the current system name and passes
   it to GeFetch.br.

   Installation involves copying both scripts to Thor/rexx, editing
   the GFPath variable to point to your copy of GoFetch! and setting
   FetchReadme to 1 if you want GoFetch! to get the Aminet readme files too


For additional information, please see the AREXX script.
```

1.25 aminetfetch.rexx

```
                         AminetFetch.rexx
                       by Olivier Fabre


 Insert all files listed in an Aminet INDEX or RECENT file
 into GoFetch! profiles list.
 GoFetch! will be run if necessary.

 Template :
   FILE=F,GETREADMES=GR/T,LOCALDIRS=LDS/T,AMINETSERVER=AS/K,AMINETPATH=AP/K,
    LOCALDIR=LD/K,GOFETCH=GF/K

   FILE
       Name of the file containing the list of files to be
       downloaded from Aminet (default : "AMINET_EXTRACTED").

       Format of this file :
```

```
            <filename> <dir/subdir> <whatever>
            ...

    GETREADMES/T
        YES/ON : Download readme files (default).
        NO/OFF : Download only the main file.

    LOCALDIRS/T
        YES/ON : Recreate the Aminet tree locally (default).
        NO/OFF : Put all files in the same directory (LocalDir).

    AMINETSERVER/K
        Your favourite Aminet server (default : "de.aminet.net").

    AMINETPATH/K
        Path to Aminet on the server (default : "/pub/aminet/").
        This shouldn't need to be changed.

    LOCALDIR/K
        Where to put the downloaded files (default : "DOWNLOAD:").

    GOFETCH/K
        GoFetch! location (default : "NET:GoFetch!/GoFetch!").


 Required :
    RexxDOSSupport.library (Copyright (C) 1994-1997 by hartmut Goebel)
    (Aminet:util/rexx/rexxdossupport.lha)

For additional information, please see the AREXX script.
```

1.26 aminetextract.rexx

```
                        AminetExtract.rexx
                         by Olivier Fabre

 Extract selected lines out of an Aminet INDEX based on date and/or type

 Template :
    IF=INDEXFILE,EF=EXTRACTFILE,FW=FIRSTWEEK/N,LW=LASTWEEK/N,NT=NOTYPE/S,T=TYPE/M

    INDEXFILE
        INDEX file name.
        Default : "INDEX".

    EXTRACTFILE
        Destination file name.
        Default : "AMINET_EXTRACTED".

    FIRSTWEEK/N
        Oldest week, numbered from 0 (current week) to x (x weeks old).
        Default : 52

    LASTWEEK/N
        More recent week, numbered from 0 (current week) to x (x weeks old).
```

```
      Default : 0

   NB : "current week" means "the week the INDEX file was created" of course...

  TYPE/M
      The type of files to extract, in the form "dir/subdir" or "dir".
      Default : all types.

  NOTYPE/S
      If set, the files of the given type will NOT be included in the list.

 Requires :
   RexxDOSSupport.library (Copyright (C) 1994-1997 by hartmut Goebel)
   (Aminet:util/rexx/rexxdossupport.lha)

For additional information, please see the AREXX script.
```

1.27 yam2gofetch.rexx

```
                                        yam2gofetch.rexx
                         by Stephen Bridges


  Is a script to batch download files from Aminet Recent files.

  It was written to work from Yam using Aminet Recent e-mails
  but any Recent file will work. As Yam has it's own built in
  text clipping (to the clipboard) it is simplier to use from
  within Yam. Powersnap or the Snap facility within MCP is
  required to paste the clipped text into the requester. It is
  best to clip the text in, as the script relies on the spacing
  between the file name and the path to determine if the text
  is a valid path and file. The script then makes up a batchfile
  which is sent to the profile list in GoFetch!. Or the batchfile
  can be saved for later use with
              Batch2GoFetch!.rexx
                 .

  Requirements

  PowerSnap          Aminet.....util/cdity/PowerSnap22a.lha
  rexxreqtools.library Aminet.....util/rexx/RexxReqTools.lha
  reqtools.library    Aminet.....util/libs/ReqToolsUser.lha

  Adjustment

  Change the paths to suit your set-up.

  gofetch_path = 'AmiTCP:GoFetch/GoFetch!' Line 21
  miami_path =   'AmiTCP:Miami/Miami'      Line 22
  settings =     'Miami:Miami.default'     Line 23

  Enter the name of your nearest Aminet server aand the path to the site.

  site =         'ftp.demon.co.uk'         Line 25
  dir_address =  '/pub/mirrors/aminet/'    Line 26
```

Set the path to the where you want the batchfile saved.

batch_path =   'Yam:gofetch.batch'   Line 28

Yam Setup

Just enter it as one of Yams Arexx scripts in the configuration,
without "open window" and "wait for termination".

1.28 batch2gofetch.rexx

                                 batch2gofetch.rexx
                        by Stephen Bridges

Is a script to batch download files from Aminet from a file
built up by
              Yam2GoFetch!.rexx
                 .

Requirements

rexxreqtools.library Aminet.....util/rexx/RexxReqTools.lha
reqtools.library   Aminet.....util/libs/ReqToolsUser.lha

Adjustments

Change the paths to suit your set-up.

gofetch_path = 'AmiTCP:GoFetch/GoFetch!' Line 22
miami_path =   'AmiTCP:Miami/Miami'   Line 23
settings =   'Miami:Miami.default'   Line 24

Enter the name of your nearest Aminet server.

site =   'ftp.demon.co.uk'   Line 25

Set the path to the batchfile the same as you set it in

              Yam2GoFetch!.rexx
                 .

batch_file =   'Yam:gofetch.batch'   Line 27

1.29 Products

                                 Other Products

Other products from Zeus Developments include:

```
Zeus BBS - Verison 1.2
An Amiga BBS package

SlamMail
A Fidonet Mailer & Tosser
```

1.30 prereg

Registering

Payments may be made by cash or cheque in the following currencies. Please make cheques payable to Zeus Developments and send to the address shown below.

```
Australia               = $24.00
Austria                 = Sch120.00
Belgium                 = Fr613.00
Canada                  = $23.00
Denmark                 = Kr113.00
France                  = Fr100.00
Germany                 = DM30.00
Greece                  = Dr5010
Italy                   = Lir30000.00
Japan                   = Y1610.00
New Zeland              = $30.00
Norway                  = Kr123.00
Poland                  = Z60.00
Portugal                = Esc3045.00
Republic of Ireland     = £12.00
Spain                   = Pst2527.00
Sweden                  = Kr130.00
United Kingdom          = £8.00
United States           = $16.00
```

```
Zeus Developments
12 Burnbeck Place,
Heighington,
Newton Aycliffe
Co Durham.
DL5 6RW
England, UK.
```

Registration Form for Go Fetch!

I would like to register Go Fetch! at the registration price of £8ukp, (or equivalent price in my local currency).

Name: _____

Address:_____

_____

_____

_____

Post/ZIP Code_____

Country_____

Your E-Mail Address:_____
(Your keyfile will be sent to this address)

Signature_____
(If handwritten)

1.31 Mailing List

Mailing List

If you would like to subscribe to the Go Fetch! mailing list to discuss Go
Fetch! with other users and the author, or to receive information on new
updates, please follow these instructions:.

NB This List Server is still under construction, it may not always behave
itself.

Mailing List Subscription Address: GoFetch@bleach.demon.co.uk
Permitted Languages: English

Commands should be sent in the SUBJECT like of messages to the above
address. Mail with commands in the subject will NOT be sent to the rest
of the list.

Mailing List Commands:

Subscribe
    Adds the senders E-Mail address to the list. If you are already subscribed
    then Rhyme will ignore this command and pass the mail onto the rest of the
    list

Remove
    Removes the sender from the list. MUST be the only word in the
    subject

ListUsers
    Replies with the list of people on the list

Help
    Sends the current Help File for the list. MUST be the only word in
    the subject.

Problems should be sent to dbny@bleach.demon.co.uk

1.32 faq

                                        FAQ

Q1. Go Fetch! says it can't open the preferences or profiles file, even though
    it exists in the right location, why?

A1. You may be running a background disk compression utility such as
    DiskExpander or stacker. Some disk compression utilities to not allow files
    to be created in READWITE mode. Please install it on an uncompressed
    partition.

Q2. Are you going to add an FTP browser type window?

A2. Probably not, as this would be turning Go Fetch! into a conventional FTP
    client which is not my intention.

Q3. How do I create an FTP profile from a complete URL?
    (e.g.ftp://www.whatever.com/pub/file.lha)

A3. Use the supplied AREXX script called browser.rexx.
See
                    Arexx Scripts
                    Q4. I'm currently using Miami or Miami DX and sometimes I get the ←
                        error message
    "Error associated with the bind command". Why?

A4. The short answer is, I don't know at the moment, but I'm working on it. A
    quick fix is to quit your TCP/IP stack and restart it or waiting a minute or
    so before continuing with any profiles.

Q5. Will Go Fetch! have locale support?

A5. Yes

Q6. I had a partially downloaded file, but Go Fetch! did not resume, why?

A6. The server containing the file does NOT support file resuming.

Q7. There's a file on a web site I want, but Go Fetch! says it isn't there, even
    though my web browser does, why?

A7. Version 1.0 of Go Fetch! does not support HTTP re-direction. It will in the
    next release.

Q8. I like Go Fetch! so much, can I give you a lifetimes supply of FREE beer?

A8. Of Course! :)