

arexx

COLLABORATORS

| | | | |
|---------------|-------------------------|-------------------|------------------|
| | <i>TITLE :</i> arexx | | |
| <i>ACTION</i> | <i>NAME</i> | <i>DATE</i> | <i>SIGNATURE</i> |
| WRITTEN BY | | February 12, 2023 | |

REVISION HISTORY

| NUMBER | DATE | DESCRIPTION | NAME |
|--------|------|-------------|------|
| | | | |

Contents

| | | |
|----------|-------------------------------|----------|
| 1 | arexx | 1 |
| 1.1 | The ARexx Interface | 1 |
| 1.2 | ascii | 1 |
| 1.3 | beep | 2 |
| 1.4 | binary | 2 |
| 1.5 | bufferize | 2 |
| 1.6 | cd | 2 |
| 1.7 | close | 2 |
| 1.8 | delete | 2 |
| 1.9 | flush | 2 |
| 1.10 | get | 3 |
| 1.11 | getreply | 3 |
| 1.12 | help | 3 |
| 1.13 | hide | 3 |
| 1.14 | iconify | 3 |
| 1.15 | lcd | 4 |
| 1.16 | log | 4 |
| 1.17 | loglevel | 4 |
| 1.18 | logsize | 4 |
| 1.19 | mget | 5 |
| 1.20 | mkdir | 5 |
| 1.21 | mput | 5 |
| 1.22 | open | 5 |
| 1.23 | pager | 5 |
| 1.24 | printdir | 6 |
| 1.25 | put | 6 |
| 1.26 | quiet | 6 |
| 1.27 | quit | 6 |
| 1.28 | quote | 6 |
| 1.29 | rename | 6 |

| | |
|---|---|
| 1.30 show | 6 |
| 1.31 site | 7 |
| 1.32 timeout | 7 |
| 1.33 verbose | 7 |
| 1.34 version | 7 |
| 1.35 ReadArgs Command Templates | 7 |

Chapter 1

arexx

1.1 The ARexx Interface

The ARexx Interface

The ARexx interface allows automation of Gui-FTP's features using a commands set that resemble that offered by ncftp. Gui-FTP can be run more than once and each instance will be given its own ARexx port. The port is called GUIFTPn where 'n' is the port number starting at 1, i.e. the first instance of the program will have a port called "GUIFTP1". The actual port name is shown in the "About" requester.

The commands implemented are summarised here.

ASCII BEEP BINARY BUFFERSIZE
CD CLOSE DELETE FLUSH
GET GETREPLY HELP HIDE
ICONIFY LCD LOG LOGLEVEL
LOGSIZE MGET MKDIR MPUT
OPEN PAGER PRINTDIR PUT
QUIET QUIT QUOTE RENAME
SHOW SITE TIMEOUT VERBOSE
VERSION

The following sections explain each of these commands. In the description, the first line gives a summary of the command using AmigaDos `ReadArgs()` format.

1.2 ascii

ASCII

Switches to ASCII transfer mode. Files transferred after this command is issued will be treated as text files and carriage return (CR) and line feed (LF) codes will be correctly translated. See Choosing the Transfer Mode for additional information.

1.3 beep

BEEP ALL/S

The action of this command depends on your system preferences. It will either sound an audible bell, or flash the screen, or do both. This can be useful to alert yourself when a download has finished.

1.4 binary

BINARY

The opposite of ASCII, after this command, files will be transferred in binary mode. There will be no translation of CR and LF codes. See Choosing the Transfer Mode for additional information.

1.5 buffersize

BUFFERSIZE SIZE/A/N

Sets the size of the transfer buffer. It is exactly the same as the tootype of the same name.

1.6 cd

CD DIR/A

Changes directory on the host.

1.7 close

CLOSE

Closes the connection to the host.

1.8 delete

DELETE FILE/A/M

Deletes a directory or file on the host. It can take multiple arguments if multiple files are to be deleted but it currently does not support wildcards. The host may require directories to be empty before deleting them.

1.9 flush

FLUSH

Flushes the directory cache and releases memory allocated to it.

1.10 get

GET FILE/A,NEWNAME

Gets a file from the host, optionally giving it a new name on the local machine.

1.11 getreply

GETREPLY

This command is intended for advanced FTP users who use the QUOTE and SITE commands to directly issue FTP commands. Getreply is used to resynchronise after executing a command using QUOTE or SITE when the replies sent by the host get out of step with the commands executed. If there are no replies outstanding, Gui-FTP will hang until the timeout interval elapses or it is interrupted with a CTRL-C signal.

1.12 help

HELP COMMAND

Displays the commands available at the ARexx port or if the command name is given as a parameter gives help for that command

1.13 hide

HIDE

Closes the window and places a small icon on the desktop. The program remains active and continues to accept ARexx commands. Double clicking the icon or using the SHOW ARexx command will re-open the window. ICONIFY is a synonym for HIDE.

1.14 iconify

ICONIFY

Closes the window and places a small icon on the desktop. The program remains active and continues to accept ARexx commands. Double clicking the icon or using the SHOW ARexx command will re-open the window. ICONIFY is a synonym for HIDE.

1.15 lcd

LCD DIR/A

If the named local directory exists, it is made current, otherwise the current directory remains unchanged.

1.16 log

LOG TEXT/M

Writes a string to the session history. The string will always be written to the log file. Whether or not it appears in the session history in the main window depends upon the setting of the loglevel.

1.17 loglevel

LOGLEVEL LEVEL/A

Sets the severity of messages that are displayed in the session history portion of the main window. Regardless of this setting all log messages will be written to the log file, even if not displayed in the window. The LEVEL may be one of:

1. DEBUG - all messages received from the host will be written to both the log file and the history display in the main window. If the program has been compiled to contain debugging information then this will also be written.
2. INFO - all messages with a severity level of INFO (informational) or above are written. Debugging information is suppressed. This is the default unless the DEBUG tooltype has been defined.
3. ERROR - only messages with a severity of ERROR or above are written. DEBUG and INFO messages are suppressed.

Messages that are not preceded by a severity code (e.g. "[INFO]" or "[ERROR]", etc.) are replies from the host and are written to the log regardless of the LOGLEVEL setting.

1.18 logsize

LOGSIZE SIZE/N/A

Sets the number of entries that will be displayed in the session history. If the number specified is smaller than the number actually allocated, then the oldest entries will be removed until there are the specified number left.

Specifying zero means that there will be no limit on the number held, other than the amount of memory your computer has, of course.

1.19 mget

MGET FILE/M/A

Get multiple files from the host. Files will be stored on your computer with the same name as they have on the host, there is no provision for renaming them with this command.

1.20 mkdir

MKDIR DIR/M/A

Creates a new directory or directories on the host.

1.21 mput

MPUT FILE/M/A

Uploads multiple files from your computer to the host.

1.22 open

OPEN SITE/A,USER/K,PASSWORD/K,ACCOUNT/K,DIR/K,LISTCMD/K,FILTER/K

Opens a connection to a site. If there is already a connection to another site, that connection will be closed before attempting to open the new connection. Gui-FTP will attempt to find the first entry in the file "Gui-FTP.rc" that wholly or partially matches the site name given. The match need not be exact, it can be any sequence of characters from anywhere in the site name. For example if there is an entry in this file for "ftp.wustl.edu" then any of the following would find this entry (assuming that this is the first entry in the file):

ftp.,

wustl

ftp.wustl.edu

If any of the optional keywords are given, their values override those obtained from the "RC" file.

1.23 pager

PAGER PROGRAM/A

Sets the program that will be used to view files.

1.24 printdir

PRINTDIR

Prints the list of files displayed in the remote directory to the log file.

1.25 put

PUT FILE/A,NEWNAME

Uploads a file to the host, optionally renaming in the process.

1.26 quiet

QUIET

The opposite of VERBOSE, only significant messages and command replies will be shown in the session history and the log file.

1.27 quit

QUIT

Exits Gui-FTP. If the program is still connected to a site, the connection is closed before the program quits.

1.28 quote

QUOTE COMMAND/F

For advanced users. This passes an FTP command directly to the host.

1.29 rename

RENAME FROM/A,TO/A

Renames a file or directory on the host. You must be authorised to rename files for this command to succeed.

1.30 show

SHOW

Reopens the window and removes the icon from the desktop. It is used after a HIDE command.

1.31 site

SITE COMMAND/F

Passes a site specific command to the host. Site specific commands are not defined in the FTP specification and as such not all sites support this command.

1.32 timeout

TIMEOUT VALUE/N

Sets the timeout value. It is identical to the tooltype of the same name.

1.33 verbose

VERBOSE

Switch Gui-FTP into "verbose" mode - all commands sent to the host and the replies received will be shown in the session history. The default mode is **QUIET**.

1.34 version

VERSION

Show Gui-FTP's version, build number and build-date in the session history.

1.35 ReadArgs Command Templates

ReadArgs Command Templates

ReadArgs command templates are the standard way of specifying the arguments accepted by a program. In general they are specified as:

command [argument-template]

where, command is the name of the program to run and argument-template is a description of the arguments accepted. The template comprises a number of argument names and optionally followed by modifiers. Modifiers are specified as a single alphabetic character preceded by a "/" and they follow immediately after the option name. Multiple modifiers may be specified by using multiple "/"s. The modifiers are:

/A Required. This parameter must be given a value otherwise an error is flagged.

/F Rest of line. The whole of the rest of the line.

/K Keyword. A value is assigned only if the keyword is entered in the

command line.

/M Multiple strings. Any strings remaining after all other modifiers are satisfied (except /F) are assigned to this parameter.

/N Number. The argument must be a number.

/S Switch.

/T Toggle

For a complete description of these modifiers, refer to the description of the ReadArgs() function in the ROM Kernel Reference Manual - Includes and Autodocs, published by Addison Wesley.
