Wikiprint Book

Title: Documentation/ARexxAPI

Subject: YAM - Yet Another Mailer - Documentation/ARexxAPI

Version: 27

Date: 29/05/15 22:32:52

Table of Contents

Languages: 3
Reference - ARexx interface 3
List of all the ARexx commands sorted by function 4
Commands supplied by MUI 4
Address book handling 4
Folder wide operations 4
GUI handling 5
Message/Folder selection 5
Message manipulation 5
Network related 5
Request info from user 5
Writing new email 6

Languages:

- **English**
- Español
- Français

Reference - ARexx interface

With the help of ARexx scripts, you can add new functions to YAM or let it do things automatically.
YAM offers a set of commands (listed in this documentation both by name ? and by function ?) which can be called through the **YAM** ARexx port. They are explained along the following subchapters using the following format:

NAME

The name of the command, a short description of what it does and the YAM version where the command was implemented.

TEMPLATE

Arguments and options accepted by the command. The template uses special characters that indicate the particular type of argument expected, following the AmigaDOS template style:

`/A` The parameter is compulsory
`/K` The parameter must be preceeded by the keyword
`/N` Numerical argument or result
`/M` Argument or result is a list of zero or more elements
`/S` The parameter works as a switch; the switch is in use when the parameter is given

FUNCTION

Describes what the command will do.

INPUTS

Describes in detail the parameters accepted by the command. Be careful when passing arguments containing spaces; for instance,

```
sub = 'Hello World'
'WRITESUBJECT' sub
```

won't work. It must be written as

```
'WRITESUBJECT "'sub'"'
```

or

```
'WRITESUBJECT "Hello World"'
```

Please note that because of the internal use of the ReadArgs() function, the ARexx Host requires to escape certain special characters like a newline (0x0a) or escape character (0x1b) if you want to have it included in your final string or otherwise it is stripped by the ReadArgs() function.

This means that you have to use the following escape sequences in your provided strings:

`*N` substitutes to 0x0a (newline \n)
`*E` substitutes to 0x1b (escape)
`!**` substitutes to *
`*` substitutes to "

For example the following command would write a string to a texteditor containing a newline:

```
'WRITEEDITOR "TEXT Hello Joe,*N I would like to meet you."'
```

If a parameter represents a file name you should keep in mind that file names on AmigaOS might contain spaces or other special characters which will cause ARexx to treat the name as multiple words instead of a single word. Hence a file name should **always** be surrounded by quotes, no matter if it contains spaces or not. Scripts can very easily be tested for this case by letting them handle a file in the Amiga RAM disk, i.e. "RAM Disk:/foo/bar".

RETURNS

Returned info to be expected from the command. Commands may return results in three different kinds; let's look at these examples which use the FOLDERINFO command:

```
FOLDERINFO
-> RESULT = "0 Incoming incoming 10 2 4 23030 1"

FOLDERINFO VAR fi
-> fi = "0 Incoming incoming 10 2 4 23030 1"

FOLDERINFO STEM fi.
-> fi.number = 0
  fi.name = "Incoming"
  fi.path = "incoming"
  fi.total = 10
  fi.new = 2
  fi.unread = 4
  fi.size = 23030
  fi.type = 1
```

Another example which returns a result of type `/M`:

```
ADDRFIND STEM found. "Marcel Beck" NAMEONLY
-> found.alias.count = 2
  found.alias.0 = "Mars"
  found.alias.1 = "mbe"
```

WARNING

Any sort of vital information you should be aware of when using this command.

NOTES

Various notes about the command.

EXAMPLE

A fragment of ARexx code to illustrate the usage of the command.

BUGS

Problems that are known or have already been fixed with this command.

SEE ALSO

Links to other related commands.

List of all the ARexx commands sorted by function

Commands supplied by MUI

HELP - Return a list of all the supported ARexx commands
HIDE - Iconify YAM
INFO - Return information about the program
QUIT - Terminate YAM
SCREENTOBACK - Bring the YAM screen to the background
SCREENTOFRONT - Bring the YAM screen to the foreground
SHOW - Uniconify YAM

Address book handling

ADDRDELETE - Remove an entry from the address book
ADDREDIT - Fill the fields of the current address book entry
ADDRFIND - Search the address book for one or more entries
ADDRGOTO - Select an address book entry by alias
ADDRINFO - Return information about an entry referred by alias
ADDRLOAD - Load an address book by filename
ADDRNEW - Create a new entry in the address book
ADDRRESOLVE - Resolve an alias or name
ADDRSAVE - Save the address book under a given filename

Folder wide operations

FLUSHINDEXES - Flush the index of all folders
FOLDERINFO - Return information about a given folder
GETFOLDERINFO - Return information about the current folder
MAILFILTER - Filter messages in the current folder
MAILUPDATE - Update the index of the current folder
NEWMAILFILE - Return a free message slot for a given folder

GUI handling

APPBUSY - Switch YAM to busy mode
APPNOBUSY - Bring YAM out of busy mode
LISTFREEZE - Forbid graphical updates to a list
LISTUNFREEZE - Permit graphical updates to a list
READCLOSE - Close the active Read window
RESTART - Restart YAM

Message/Folder selection

FINDMAIL - Find a mail in all folders given a MessageID
GETSELECTED - Get the position of the messages selected in the current folder
LISTSELECT - Select messages in the current folder
SETFOLDER - Select the current folder
SETMAIL - Select the current message in the current folder by number
SETMAILFILE - Select the current message in the current folder by filename

Message manipulation

GETMAILINFO - Return information about the active message
MAILARCHIVE - Move the selected messages to another folder
MAILBOUNCE - Bounce the current message
MAILCHANGESUBJECT - Change the subject of the selected message(s)
MAILCOPY - Duplicate the selected message(s)
MAILDELETE - Delete the selected message(s)
MAILEDIT - Edit the current message
MAILEXPORT - Export the selected message(s) into a file
MAILIMPORT - Import messages from a file into the current folder
MAILINFO - Return information about a given message
MAILMOVE - Move the selected messages to another folder
MAILREAD - Read the current message
MAILREPLY - Reply the currently selected messages
MAILSTATUS - Change the status of the selected message(s)
READINFO - Get info about the attachments in the message being read
READPRINT - Print an attachment
READSAVE - Save an attachment
SETFLAG - Set the PERmanent or VOLatile flag of a message

Network related

GETURL - Download a file from the WWW based on a given URL
ISONLINE - Detect whether YAM is currently transfering email
MAILCHECK - Download new email
MAILSEND - Send the messages in the **Outgoing** folder
MAILSENDALL - Send all the messages in the **Outgoing** folder

Request info from user

GETCONFIGINFO - Return information about the current user
REQUEST - Request a choice from the user
REQUESTFILE - Request the user to select a file
REQUESTFOLDER - Request the user to select a folder
REQUESTSTRING - Request the user for a string
USERINFO - Return information about the current user

Writing new email

MAILFORWARD - Forward the selected messages
MAILWRITE - Open a new Write window to compose a new mail
WRITEATTACH - Attach a file to a new message
WRITEBCC - Fill the BCC: header of a new message
WRITECC - Fill the CC: header of a new message
WRITEEDITOR - Pass an ARexx command to the internal editor
WRITEFROM - Fill the FROM: header of a new message
WRITEIDENTITY - Choose the user identity to be used when composing a new mail
WRITELETTER - Load a file into the internal text editor
WRITEMAILTO - Fill the TO: header of a new message
WRITEOPTIONS - Set the options of a new message
WRITEQUEUE - Move a new message to the **Outgoing** queue
WRITEREPLYTO - Fill the REPLY-TO: header of a new message
WRITESEND - Send a new message
WRITESUBJECT - Fill the SUBJECT: header of a new message
WRITETO - Fill the TO: header of a new message