

MAIN

COLLABORATORS

	<i>TITLE :</i> MAIN		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		December 2, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	MAIN	1
1.1	ArCo MAIN WINDOW HELP	1
1.2	The Program Menus	1
1.3	Reading the souce file	2
1.4	Editing the souce file	2
1.5	Executing the souce file	2
1.6	New Cli	2
1.7	Switching the GUI to Iconify	3
1.8	How to Disapoint the punter...	3
1.9	Quitting ArCo	4
1.10	Load a Preference file	4
1.11	Saving a preference file to default	4
1.12	Saving a preference file	4
1.13	Executing an ArCo Arexx Macro	5
1.14	Buttons & Keyboard shortcuts for the main Window	5
1.15	ArCo Preferences	7
1.16	Amending the tooltypes	8

Chapter 1

MAIN

1.1 ArCo MAIN WINDOW HELP

ArCo HELP © M A SEWELL '94

AmigaGuide, AmigaGuide.info, amigaguide.library, WDisplay, WDisplay.info
 © Copyright 1991-93 Commodore-Amiga, Inc. All Rights Reserved.
 Reproduced and distributed under license from Commodore.

Please select any of the topics listed below.

```
@{ " Preferences                " link prefs }
@{ " The Program Menus          " link menus }
@{ " Buttons & short cut keys   " link shortcuts }

@{ " ArCo Manual                " link ARCO:ArCo_Manual.guide/main }
```

AMIGAGUIDE SOFTWARE IS PROVIDED "AS-IS" AND SUBJECT TO CHANGE;
 NO WARRANTIES ARE MADE. ALL USE IS AT YOUR OWN RISK. NO LIABILITY
 OR RESPONSIBILITY IS ASSUMED.

1.2 The Program Menus

The menu items available:

```
Project          Preferences  Arexx

@{ " Read Source      'R' " link readsrc }  @{ " Load          'L' " link loadpref }  @ ←
  { " Execute Macro  'M' " link execmacro }
@{ " Edit Source      'E' " link editsrc }  @{ " Save DEF       'S' " link savepref }
@{ " Execute Source   'X' " link execsrc }  @{ " Save as       'A' " link saveprefas }
=====
@{ " New Cli          'C' " link shell }
@{ " Iconify          'I' " link iconify }
@{ " About            '?' " link about }
```

```
=====
@{ " Quit          'Q' " link quit }
```

1.3 Reading the souce file

This menu item will invoke the text viewer you have specified in the
@{ " Preferences " link prefs } and allow you to read the source file
that you have chosen to compile.

Menu short - cut : R

1.4 Editing the souce file

This menu item will invoke the editor you have specified in the
@{ " Preferences " link prefs } and allow you to edit the source file
that you have chosen to compile.

Of use when you have RUN the compiled file and discovered errors.

Menu short - cut : E

1.5 Executing the souce file

This will cause the source file to be executed.

NOTE:

Execution will be via the 'RX' command rather than ArCo's internal
Arexx system so that the src file is run in a "standard" environment.

Menu short - cut : X

1.6 New Cli

This will induce a Shell window.

So long as Arexx is available, This shell will in fact be a command
shell.

This mean that you can address ArCo's arexx commands directly.
You can also start further Arexx scripts or use dos commands etc.

if for some reason Arexx is NOT initialised, ArCo will revert to using
a standard type shell,
Whether this is commodores shell, Or one of your own choice such
as WShell or csh, depends on your @{ " Preferences " link prefs } setting for the
'SHELL' tooltype.

NOTE:

The command shell is in fact implemented as an Arexx script and can be edited as such.

You may wish to compile it and use it as a separate tool, (Note that it ADDRESS's ArCo's port, You may want to remove this line if you wish to use it as a standalone tool!)

The Command Shell opens at the BOTTOM of your screen, if you use a PAL system, you will find your shell window beneath the main Window allowing you to see everything that's going on.

If you use NTSC, It will still Open at the bottom of the screen, though it will obscure the bottom row of buttons.

(if you are using interlace, the shell window may well appear to be divorced from ArCo!!)

To get the template for any of the Arexx commands, Simply enter Help <command>

Menu short - cut : S

If you are using a floppy based system, you may find it a bit tedious waiting for the CmdShell to load and sort itself out, You can cut down on this time by editing the CmdSHELL.acrx file and removing the help system, Only do this on a backup!

OR

Rename the file CmdSHELL.acrx to something else, Then rename CmdSHELL.crip to CmdSHELL.acrx. (This effectively does the above for you)

1.7 Switching the GUI to Iconify

This shuts down the main window, releasing some memory back to the system (not much!) and then sits quietly at the top of the screen.

Just click on the Close button to revert to the main Window again.

Menu short - cut : I

1.8 How to Disappoint the punter...

This simply opens a window that doesn't really convey any information at all about the program, Just gives the authors name and address.

Once you've seen it, you will be suitably disappointed and will not bother again!

Menu short - cut : ? (As if your worried)

Only an egotistical programmer would Write a great deal about himself, or indeed add an Arexx function to his program that would allow the conveying of said information.

And before you search the Arexx Command index, Yes, Its there!
(Sad, isn't it?)

1.9 Quitting ArCo

This quits the program, You will be presented with a requestor prior to quitting depending on whether you have set the `@{ " preferences " link prefs }` `'QUIT_REQUEST'` tooltype.

If set to FALSE, no requestor appears and ArCo simply exits.

Menu short - cut : Q

1.10 Load a Preference file

This will present a file requestor and allow you to load a preference file which contains settings for ArCo, such as Link libraries, output type etc.

This preference file must have been created previously using `@{ " Save as " link ↔ saveprefas }` from the menu or the Arexx command `equivelent`.

A file extension of `".ac_prefs"` is expected.

1.11 Saving a preference file to default

All the ArCo settings will be saved out to the default preference file. These include the list of link libraries, trace options, interrupt options plus the items that effect the finished file such as Write icon, version and run as well as verbose.

The preference file to be used as default is named in the icon tool Tool Types `@{ " preferences " link prefs }`.

1.12 Saving a preference file

All the ArCo settings will be saved out to the file chosen using the `filerequestor`.

A file extension of `".ac_prefs"` is expected, It is appended if not supplied

1.13 Executing an ArCo Arexx Macro

Using the `filerequestor`, choose an ArCo Arexx macro you wish to execute. A file extension of `".acrx"` is used for all ArCo scripts. (although it can be omitted, It will help for identification purposes)

All macro's should be stored wherever you have `REXX:` assigned (usually the `S:` directory) *This does NOT apply to `CmdSHELL.acrx` which resides in `ARCO:Support/`

1.14 Buttons & Keyboard shortcuts for the main Window

You will already have noticed that some of the words have a letter underlined, This is the shortcut key for that particular button or gadget,

e.g.

`'T'` will change the type of file that is written out by ArCo after compilation.

There are four types, `EXECUTABLE`, `LIBRARY OBJECT`, `LINKABLE OBJECT` and also `SCRIPT`.

`EXECUTABLE` = Doesn't really need explaining, A stand alone tool that can be run from `Cli` or `WorkBench` with full argument parsing (`inc. Multiple selection from WorkBench`).
(Arexx will still need to be available)

`LIBRARY OBJECT` = This produces a linkable object which is then added to the link library of your choice, That library may be determined within the source file by use of the keyword `#linklib` at the head of the script(Must include full path e.g. `ArCo:LIB/MyLIB.lib`), or by selecting a link library from the list shown in the main Window. Any link library declared within a source file takes precedence.
if no link library is declared or selected, A file requestor will ask for the destination library.(This requestor will persist until a library name is provided)
(object files do NOT have to be added to a library at the time of compilation, You may of course use the following option and then use the seperate utility `LIBRARIAN`.

`LINKABLE OBJECT` = Produces an object module that can be linked with C or assembler programs.
This object file may also be added to a link library at a later date using the `LIBRARIAN` utility.

`SCRIPT` = Why compile an Arexx script into another script ?
This produces a standard Arexx script, BUT will add any of the options to the file that you have chosen, such as interrupts or trace options.
Any declared external functions will also have been added.
This provides a way to test your scripts if you use quite a few routines that you keep in a link library.

(external functions are declared near the top of the source file with the keyword #extern)

'S' will activate the Source file string gadget.

Using shift+S will bring up a filerequestor (The same as clicking on the GetFile gadget to the right of the string gadget)

'O' will activate the Object file string gadget.

Using shift+O will bring up a filerequestor (The same as clicking on the GetFile gadget to the right of the string gadget)

'E' will activate the Executable file string gadget.

Using shift+E will bring up a filerequestor (The same as clicking on the GetFile gadget to the right of the string gadget)

'A' will enable the string gadget underneath the list of link libraries, allowing you to add a link library to the list.(You can use the GetFile gadget to the right of thes buttons to choose the library you wish to add)

'D' will delete the selected link library from the list.

'V' will toggle the Version gadget, When selected, The program name, your name and a unique verion numer is inserted into the finished executable file.

The actual version number is derived by the date and time of compilation.

'B' will toggle the Verbose gadget, When selected, You are kept informed throughout the compilation with the information showing in the 'ArCo stasis' window.

NOTE: On small files, The compiler will finish so quickly that you will not have a chance to see these messages.

'R' will toggle the Run gadget, Once you have selected RUN, a further button at the bottom left will become accessable, (Arguments).

RUN will execute the program once you've compiled it, By clicking on the Arguments button, You may choose how the finished program is executed, either Cli or WorkBench, You may also choose any arguments that you may wish to pass to your program at the point of execution.

'I' will toggle the icon gadget, If selected, ArCo will create icons for all files written out.

The rest of the buttons do NOT have any Keyboard short-cuts:

At the bottom of the left half of the main Window, There are three buttons which bring up the Option requestors, These are @ { " Trace " link ARCO:SUPPORT/ ↵ traceopts.guide/main }, @ { " Interrrupts " link ARCO:SUPPORT/interrupts.guide/ ↵ main } and @ { " Arguments " link ARCO:SUPPORT/Args.guide/main }. (The Arguments gadget is ↵ only enabled if RUN has been selected)

At the bottom of the right half of the main Window there are only two buttons, Albeit the most important two... These are COMPILE and QUIT. Compile will initiate the compile process, taking into account any trace or interrupt options, or external functions declared within the source file.

Quit does just that (depending on whether you have changed the setting

```
for @{ " QUIT_REQUEST " link prefs } )
```

The blue rectangle above the Compile & Quit button is the stasis window, Any information is routed through here.

1.15 ArCo Preferences

There are two types of preferences utilised by ArCo, Operating preferences and preset preferences.

The operating preferences are simply a saved operating state, This type of preference may be saved and loaded from the @{ " menu " link menus } strip.

All the preset preferences are set by @{ " amending the tooltypes " link ↵ tooltypes } found in the icon of ArCo.

The tool types available are:

'AUTHOR'

This is YOUR name, do NOT use any punctuation marks!

This string is inserted in the executable file if "Version" is enabled.

The Dos command Version does not like punctuation marks.(if encountered, it simply terminates the version string at that point)

Options : Any

Default : None

'TEMP_FILE'

This is the file that ArCo will use as a scratch file during compilation.

Options : Any

Default : T:arcoJUNK

'SHELL'

This will allow you to use your own choice of shell instead of Commodores. It would be best to use the NewShell command to induce your shell,Complete with path.

NOTE: This is only invoked if Arexx is not initialised.(The CmdSHELL takes priority)

Options : Any

Default : Commodore's NewShell

'QUIT_REQUEST'

This is a BOOLEAN value, ArCo will enquire if you are sure that you wish to quit unless this is set to FALSE, OFF or NO.

Options : TRUE or FALSE, (Will accept YES or NO, ON or OFF)

Default : TRUE.

'DEFAULT_PREFS'

This is the preference file you wish to use as default.

Options : Any

Default : ArCo:support/default.ac_prefs

'SOURCE_DIR'

This is the directory that the filerequestor will first Open when you use the GETFILE button alongside the Source file string gadget.

Options : Any
Default : SYS:

`'OBJECT_DIR'`

This is the directory that the filerequestor will first Open when you use the GETFILE button alongside the Object file string gadget.

Options : Any
Default : ARCO:Objectfiles/

`'EXECUTABLE_DIR'`

This is the directory that the filerequestor will first Open when you use the GETFILE button alongside the Executable file string gadget.

Options : Any
Default : SYS:

`'LINK_DIR'`

This is the directory that ArCo will search for Link Libraries.

Options : Any
Default : ARCO:LIB/

`'TEXT_READER'`

This denotes the utility you wish to use to Read the source file.

Options : Any
default : SYS:Utilities/More

`'TEXT_EDITOR'`

This denotes the utility you wish to use to Edit the source file.

Options : Any
Default : C:Ed

`'EDIT_SCRIPT'`

This denotes the default tool that is inserted into the script icon when you compile using the script option. Use TRUE to use the editor or FALSE to use the reader.

Options : TRUE or FALSE, (Will accept YES or NO, ON or OFF)
Default : TRUE.

1.16 Amending the tooltypes

Click once on the ArCo icon, Either use 'Info' from the pull-down menu under "icon" or use the keyboard shortcut which is Right-Amiga + "i". This will bring up a window with information about ArCo.

You will see the size of the file, That it's a tool etc. You will also see a List of tooltypes, Its this that we are interested in.

Click on a tooltype of your choice, Lets say `'QUIT_REQUEST='`, It will now be shown underneath and can now be edited.

First remove the surrounding brackets, ie `(QUIT_REQUEST=)` becomes `QUIT_REQUEST=`, and immediately after the equals sign, add the VALUE you wish to use,

Lets say FALSE because I hate those requestors that stop you getting OUT!

Now `QUIT_REQUEST=` will become `QUIT_REQUEST=FALSE` and you then press return.

You do this for all tooltypes you wish to set and finally click on the save button, The next time you load ArCo, your own choice of preferences will be used.