AssignManager

Matt Francis

COLLABORATORS			
	TITLE : AssignManager		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Matt Francis	January 18, 2023	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

AssignManager

1.1 Contents

1.2 credits

ASSIGNMANAGER V1.24
Created by Matt Francis
ReqTools by Nico François
GUI created with GadToolsBox V2.0c
This program is giftware. I demand no payment for this program,
because I feel I have benifitted greatly from the generosity of other
Amiga programmers, but the spiritually-sound among you should note
that I have worked long and hard on this software. If you appreciate
(and use) my efforts, you would cheer me up a lot by sending some
money my way. Job offers are also welcome :-)
E-Mail: m.p.francis@newcastle.ac.uk
S-Mail: Matt Francis,

142 Tamworth Road,

Fenham,

Newcastle-upon-Tyne,

NE4 5AN

England

(until July 1994)

1.3 disclaimer

This software comes with no warranty, either expressed or implied. The author is in no way responsible for any damage or loss that may occur due to direct or indirect usage of this software. Use this software entirely at your own risk.

1.4 copyright

1.5 distribution

Permission is given to include this program in a public archive (such as a BBS, FTP site or PD library) providing that all parts of the original distribution are kept intact. These are as follows:

AssignManager/Assigns

AssignManager/Assigns.info

AssignManager/AssignManager.guide

AssignManager/AssignManager.guide.info

AssignManager/IMPORTANT

AssignManager/IMPORTANT.info

AssignManager/ConvertPrefs

AssignManager/ConvertPrefs.info

AssignManager/ConvertPrefs.rexx

AssignManager.info

Libs/reqtools.library

No charge may be made for this program, other than a reasonable copying fee, and/or the price of the media.

1.6 requirements

AssignManager requires Kickstart V36 or above and Nico François' ReqTools library V38 or above.

If you don't have ReqTools, the latest version of the library is supplied, which is all you need to use AssignManager. However, I would recommend getting the latest version of the ReqTools distribution, which includes useful things such as docs, a prefs editor and ARexx interface for ReqTools and, of course, the excellent RTPatch.

1.7 installation

To install AssignManager, drag the "Assigns" program into the drawer where all your other prefs editors are (usually "SYS:Prefs"). Then add the following line near the start of your User-Startup :

Assigns USE

(NOTE: If your prefs drawer is not in your DOS path, you will have to use something like "SYS:Prefs/Assigns USE" instead. This won't be a problem if you use a standard configuration.)

Once you've done this, you'll never need to add an Assign command to your User-Startup again!

A final note: when you first run AssignManager, you will

(presumably) not have an AssignManager prefs file. If this is the case, AssignManager will automatically grab the current assigns. You should delete any system or startup-sequence assigns from those AssignManager has grabbed. If you're not sure which ones to remove, clear the whole lot and start adding your own assigns from scratch.

1.8 The User-Startup file

This file is a special extension of the Startup-Sequence file that is executed when your system (re)boots. As each generation of Amigas becomes more sophisticated, the Startup-Sequence gets more complex too, and it now contains so many important commands that Commodore advise against tampering with it. In their wisdom, Commodore forsaw that people would need to add commands to the startup process, so they created the User-Startup method to allow for this. If a DOS script file called "User-Startup" exists in the S: directory, it will be executed right at the end of the Startup-Sequence, just before Workbench is started. These means that users are able to add their own Assigns or Paths and run programs and patches that can't be started via the WBStartup drawer without the need to touch the Startup-Sequence file in any way.

1.9 introduction

Please read the History section for information on changes and bug-fixes.

AssignManager is a new prefs editor which handles your custom assigns in a friendly, all-encompassing way. No more fiddling with Assign commands in User-Startups . Now you can use AssignManager to edit your list of assigns to your heart's content, and you can change them at the drop of a hat.

AssignManager functions as a true prefs editor in every way, except you can't save different presets (I didn't think this was very necessary). The only other difference is that you must add a line near the start of your User-Startup (see Installation).

1.10 Using AssignManager

If you can use Commodore's prefs editors, and if you are familiar with GadTools gadgets, you should have no problems using AssignManager.

AssignManager can be started from the Workbench or a Shell.

Format: ASSIGNS [USE] [PUBSCREEN <public screen name>]

Template: USE/S,PUBSCREEN/K

Native tool-types: PUBSCREEN=<public screen name>

When the program is invoked from the Workbench, or without the "USE" switch from a Shell, AssignManager's window will appear. The window contains a number of gadgets:

Add Top Up Down

Bottom Sort Delete Unassign

Clear Assign Type State

Path Comment Save Use

Cancel

AssignManager uses the standard prefs-editor format for its menus. The

Project menu is self-explanantory.

Edit menu Reset To Defaults

Last Saved

Restore

Grab Current

Sending AssignManager a CTRL-D will cause it to pop up. AssignManager can be killed with a CTRL-C .

1.11 Add Gadget

Keyboard shortcut = a/A

This gadget causes a new, blank entry to be added at the bottom of the assign list. You can fill in the new assign's details with the gadgets below the list-view.

See also: Delete , Unassign , Clear

1.12 Top Gadget

Keyboard shortcut = U

This gadget causes the current entry to be moved to the very top of the assign list.

See also: Up , Down , Bottom

1.13 Up Gadget

Keyboard shortcut = u

This gadget causes the current entry to move up one line in the assign list.

See also: Top , Down , Bottom

1.14 Down Gadget

Keyboard shortcut = d

This gadget causes the current entry to move down one line in the assign list.

See also: Top , Up , Bottom

1.15 Bottom Gadget

Keyboard shortcut = D

This gadget causes the current entry to be moved to the very bottom of the assign list.

See also: Top , Up , Down

1.16 Sort Gadget

Keyboard shortcut = r/R

This gadget causes the assign list to be sorted in alphabetical order of assign name.

The Utility library's Stricmp() function is used to compare the names, so the sort will be localised if you use the Locale library.

1.17 Delete Gadget

Keyboard shortcut = none

This gadget deletes the current entry from the assign list. If a DOS assign corresponding to the entry exists, it will not be cancelled.

See also: Add , Unassign , Clear

1.18 Unassign Gadget

Keyboard shortcut = none

This gadget functions as the Delete gadget, with the addition that if a DOS assign corresponding to the current entry exists, it will be cancelled.

See also: Add , Delete , Clear

1.19 Clear Gadget

Keyboard shortcut = none

This gadget causes the entire assign list to be cleared. It will not cancel any existing assigns. Exactly the same effect can be achieved using Reset To Defaults in the Edit menu.

See also: Add , Delete , Unassign

1.20 Assign Gadget

Keyboard shortcut = i/I

This string gadget is used to enter the name of an assign. When entering an assign name, you shouldn't add the trailing colon. Also, the assigns are not created as you enter them, but only when when you select Use or Save . If you enter an assign name that is already in the list, one of two things can happen: if the existing assign is a Simple or an Add assign, the assign you are editing will automatically be made an Add assign. Otherwise, a requester will appear warning you of conflicting incompatible assigns (you can't add multiple assigns to a

Path assign for example).

See also: Type , State , Path , Comment

1.21 Type Gadget

Keyboard shortcut = y (cycle forwards), Y (cycle backwards)

This gadget allows you to select one of the four different types of assign. Where applicable, they correspond exactly to the switches the DOS Assign command uses, and are Simple , Defer , Path and Add .

See also: Assign , State , Path , Comment

1.22 State Gadget

Keyboard shortcut = t (cycle forwards), T (cycle backwards)

This gadget allows you to render an assign inactive without having to delete its entry. This is useful for assigns that you occasionally need, but don't wish to have active all the time. When an assign is inactive, an asterisk will appear next to the type column in the listview. If an assign marked as inactive exists when AssignManager makes its changes to the DOS list, it will be cancelled.

See also: Assign , Type , Path , Comment

1.23 Path Gadget

Keyboard shortcut = p (activate gadget), P (open path requester)

This gadget is used to enter the path associated with an assign. You can enter these manually, but it is sometimes easier to click on the getfile gadget and choose a path using the path requester.

See also: Assign , Type , State , Comment

1.24 Comment Gadget

Keyboard shortcut = o/O

You can use this gadget to record comments on your assigns. This handy for remembering what certain assigns are used for, and by which programs.

See also: Assign , Type , State , Path

1.25 Save Gadget

Keyboard shortcut = s/S

This gadget will exit AssignManager, before which the current assign list is permanently recorded so that it will be in effect for the current session and all following sessions. Also, all active assigns will be created if they don't already exist. All assigns marked as

inactive will be cancelled if they exist.

See also: Use , Cancel

1.26 Use Gadget

Keyboard shortcut = e/E

This gadget will exit AssignManager, before which the current assign list is recorded so that it will be in effect for the current session, but will be gone the next time you reboot. Also, all active assigns will be created if they don't already exist. All assigns marked as

inactive will be cancelled if they exist.

See also: Save , Cancel

1.27 Cancel Gadget

Keyboard shortcut = c/C

This gadget will exit AssignManager, discarding the current list of assigns. No changes will be made to the DOS assign list.

See also: Save , Use

1.28 Edit » Reset To Defaults

Keyboard shortcut = Left Amiga + D

This menu item causes the entire assign list to be cleared. It will not cancel any existing assigns. Exactly the same effect can be achieved using the Clear .

See also: Clear , Delete , Unassign

1.29 Edit » Last Saved

Keyboard shortcut = Left Amiga + L

This menu item causes the current assign list to be replaced with the list that was stored when the Save gadget was last used.

See also: Save

1.30 Edit » Restore

Keyboard shortcut = Left Amiga + R

This menu item causes the current assign list to be replaced with the list as it was before any changes were made with the current invokation of AssignManager. It can be thought of as a sort of "undo" feature.

See also: Last Saved

1.31 Edit » Grab Current

Keyboard shortcut = Left Amiga + G

This menu item will append DOS's current set of assigns onto the end of the current assign list. All types of assign are grabbed, including multiple (add) assigns.

1.32 Assign Types: Simple (Lock)

These are conventional, single-directory assigns (like using the Assign command without any switches). Also called lock assigns.

1.33 Assign Types: Defer (Late)

These are equivalent to the Assign command's "DEFER" switch, and are also called late assigns. This type of assign doesn't bind immediately; the path is not locked (i.e. DOS doesn't search for it) until some reference is made to the assign. When this happens, the late assign becomes a lock assign. In simple terms, this means you can do things such as assigning to an unmounted volume without getting a requester.

1.34 Assign Types: Path

These are like using the Assign command's "PATH" switch. Path assigns never bind. Instead, the path you specify is used each time the assign is referenced. For example, having assigned "LIBS:" to "DF0:Libs", getting a directory of "LIBS:" will yield a directory of the "Libs" directory of whichever disk happens to be in the drive, no matter what volume it is.

1.35 Assign Types: Add

These are equivalent to the Assign command's "ADD" switch. Whereas the other types of assign replace any other assign of the same name, an add assign extends any lock assign to cover more than one directory. Therefore you can have several physical "Libs" directories, with one assign name referring to all of them (very useful for seperating your own library collection from the system libraries for example).

1.36 signals

If you started AssignManager synchronously from a Shell (i.e. you didn't use the Run command) you can send it a CTRL-C or CTRL-D signal just by pressing those keys. If you started the program using the Run command, or you are in another Shell window, use the Status command to find AssignManager's process number and then use the Break command to send it a signal.

If you started AssignManager from the Workbench, you will have to find its process number some other way or use a task monitor program (e.g. TaskX or TaskE) to send it a signal.

1.37 acknowledgments

Thanks to the following people:

Nico François for the ReqTools library, which makes life much more

pleasant.

Jan van den Baard for GadToolsBox, which was used to design

AssignManager's GUI.

CygnusSoft for CygnusEd, the fastest, most stable and best

text editor for the Amiga (and, from what I've seen, for any

computer!). And it still works without problems on my A1200!

Commodore for brilliant computers which the competition just can't

match.

Michael Simons and Lars (sorry, I don't know your last

name!) for their ideas and bug reports.

Martin Ramsch for several good ideas, notably the grab and unassign

features and the comment field.

1.38 history

V1.24

Changed the GUI around again and added some frills (consequently the Path string gadget is now a bit longer); added Clear gadget, made Grab gadget into an item in the Edit menu, added keyboard shortcuts for Type and State cycle gadgets, removed

duplicate keyboard shortcuts for Top and Bottom gadgets. Finally, when AssignManager is first started and it can't find a prefs file, it will automatically grab the current assigns.

V1.23

Optimised the sort routine. Added ability to mark particular assigns as inactive without having to delete them.

V1.22

Changed a few things round in the GUI (bigger ListView, neater buttons).

V1.21

Added check for duplicate when assign name is entered.

V1.20

Added grab feature, unassign feature and comment field. Consequently the appearance of the GUI has changed somewhat and the prefs file has changed due to the comment field. Also, added keyboard shortcuts for several buttons.

V1.11

Updated some of the code. Button gadgets now highlight when their keyboard shortcut is used.

V1.10

Window made wider again to accomodate some new gadgets, namely the sort gadget and the path getfile gadget (thanks to Michael Simons for these ideas).

V1.03

ListView text now formatted with RawDoFmt(), and an incidental bug was removed. Window is now a little narrower.

V1.02

Removed close gadget to make AssignManager look more like a Commodore prefs editor. Also, removed an undocumented feature (okay, bug) that caused an addressing error on <68020 machines (thanks Lars!).

V1.01

The up/down and top/bottom gadgets are now disabled when the particular operation cannot be performed (this is not a bug fix, just an enhancment). Added an ID port to prevent multiple invokations.

V1.00

First version. Wrote it in assembler after being dissatisfied with a similar program called AssignPrefs.