

AutoAssign

Grzegorz Calkowski

COLLABORATORS

	TITLE : AutoAssign		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY	Grzegorz Calkowski	March 28, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	AutoAssign	1
1.1	AutoAssign v1.12	1
1.2	Disclaimer	1
1.3	Copying	2
1.4	Introduction	2
1.5	Requirements	3
1.6	Installation	3
1.7	Usage from Shell	4
1.8	Usage from Workbench	4
1.9	The assignsfile	5
1.10	The Assign Wedge	6
1.11	To Do	6
1.12	Author	7
1.13	Acknowledgements	7
1.14	History	8

Chapter 1

AutoAssign

1.1 AutoAssign v1.12

»»» AutoAssign v1.12 «««
"Let's get rid of those damned assigns & mounts...!"

Beetware © 1995 The Beet Research

written by Grzegorz Calkowski

Table of contents:

Disclaimer	Legal information
Copying	Distribution & Copyright
Introduction	What the hell is it?
Requirements	Hard & soft required
Installation	Before using AutoAssign
Usage from Shell	For shell lovers
Usage from Workbench	Description of tooltypes
The assignsfile	Assignsfile description
The AssignWedge	If you use AssignWedge read it!
To Do	What should be done
History	Short program history
Author	Who made it?
Acknowledgements	Thank you!

1.2 Disclaimer

Disclaimer

The program executable and information above is provided "as is". No representations or warranties are made with respect to the accuracy, reliability, performance, currentness, or operation of this software and information, and all use is at your own risk. The author doesn't assume any responsibility or liability whatsoever with respect to your use of the program and information.

1.3 Copying

Copying

AutoAssign is published under "BeetWare" conception. It's very special sort of GiftWare. It means you should send a beet to the author (but not a rotten one! 8). Well, if you run short on beets send a disk or a small donation or anything else. (Have a hint: a bottle of beer would be excellent... :)

This program can be freely distributed under following conditions:

1. Everything in this distribution must be kept together and unaltered. Applies to:

AutoAssign	(the program itself)
AutoAssign.info	
AutoAssign.020	(68020+ version)
AutoAssign.020.info	
AutoAssign.doc	(plain-ascii doc)
AutoAssign.doc.info	
AutoAssign.guide	(AmigeGuide®'ed doc)
AutoAssign.guide.info	
MWB-Icons/AutoAssign.info	(program's icon in MagicWB style :)
MWB-Icons.info	
2. Copyright notices are not removed from executables and documentation.
3. No more than a nominal fee is charged to cover cost of magnetic media.

AutoAssign is Copyright (C) 1995 The Beet Research. All rights reserved.

1.4 Introduction

Introduction

Haven't you got tired of those applications which add more and more assigns

to your user-startup? Haven't you got irritated of those hundreds assigns showing in the requesters? I bet you have. Well, no more...

AutoAssign is a small resident program which automatically makes an assign when it is requested. It works somehow like DEFER assign but the assign won't show until it is referenced. It's the feature that such programs like AssignX, AssignWedge or ReqChange's AssignWedge lack of. Note, that AutoAssign isn't replacement for them - they are complementary. See The AssignWedge section.

From version 1.10 AutoAssign can also perform "auto" mounts. Effect is similar to so-called late mount (Mountlist's Mount keyword or DosDriver's ACTIVATE tooltype set to 0) but it has an important advantage - mount can be done also when a device is accessed by the volume name! And I think it is far more convinient to manage late mounts in one file (assignsfile, I mean) than in several mountlists, dosdrivers and startup-sequences. Even more! You can put all your devices to the assignsfile and forget about 'Mount' command!!! Nice, isn't ?

Although AutoAssign is small it occupies about 8 kilobytes of memory, but the goal was not to save a few bytes but to get rid of those rarely used assigns. Note, that memory usage doesn't depend on length of assignsfile - it's accessed every time, when unexisting assign is referenced. It also allows you to edit the assignsfile without quitting and re-running AutoAssign (or even worse - rebooting).

AutoAssign propely recognizes "Please insert volume..." requesters even in localized enviroment. Note, if you change language while AutoAssign is running you have to quit & rerun it.

AutoAssign is 100% legal and was thoroughly tested with Enforcer & Mungwall.

1.5 Requirements

Requirements

AutoAssigns requiers an Amiga (surprise?) with at least Kickstart 2.04 (NOTE: tested only on 3.0/3.1!, sorry :). It makes not much sense to use AutoAssign without harddisk. Memory requirements are negligible.

AutoAssigns requires reqtools.library V38+ installed in your LIBS: to show 'About' requester, but it won't panic if it's not installed (is there anybody without, really? :-).

The automount feature requires the 'Mount' command installed on your command path (it's part of Workbench and should sit in C:).

1.6 Installation

Installation

To have AutoAssign installed every boot you should drag it to WBStartup drawer. Of course you can put it anywhere else on your harddisk. In such case you are responsible to run it (for example from your startup-sequence or user-startup scripts). Note, that there is an alternative icon supplied in MagicWB style :-).

Note, that there is 68020+ version supplied. If you have 68020, 68030, 68040, or maybe 68060 based Amiga you should use it instead.

Also, you have to make assignsfile which consists all the assigns. By default it is "S:AutoAssigns".

1.7 Usage from Shell

Usage from Shell

Although I strongly recommend you to use AutoAssign from Workbench you can use it from Shell as well (for example, you can put it into your startup-sequence or user-startup). AutoAssign DOES NOT detach automatically from Shell so you should use 'run' command:

```
run >NIL: AutoAssign
```

As every other standard shell command AutoAssign recognizes the '?' what brings the options template.

Options are:

ASSIGNSFILE <path>	Specifies the path for assignsfile. It's optional. Default: S:AutoAssigns
--------------------	---

Note, that ASSIGNSFILE keyword can be omitted.

!!
IMPORTANT: the path must be absolute, not
relative (i.e. S:MyAssigns not S/MyAssigns).
!!

QUIET	Switch - if specified the about requester won't be shown.
-------	--

1.8 Usage from Workbench

Usage from Workbench

When launched from WB AutoAssign looks for following tooltypes:

ASSIGNSFILE=<path>	Specifies the path for assignsfile. It's optional. Default: S:AutoAssigns
	!! IMPORTANT: the path must be absolute, not relative (i.e. S:MyAssigns not S/MyAssigns). !!
QUIET	If specified the about requester won't be shown.

1.9 The assignsfile

The assignsfile

The assigns must be in pure ASCII format. Take a look at following example:

```
; AutoAssigns & AutoMounts

DPaint:      Work:GFX/DPaint
SC:          Work:SAS-C
LIB:         SC:LIB SC:LIB2/ Development:store/LIB
Toaster:     Work:Raytracing/LightWave
3d:          Data:3d                ; for LightWave

"My Game:"   "Work:Games/My Beloved Game"

FF0:         mount FF0:
Xmodules:    mount XH1: from DEVS:Mountlist.xpk
```

Any string beginning with a semi-colon (;) is considered to be a comment and is ignored. Also blank lines or any leading or trailing blanks are ignored.

Each line describes an assign or a device to mount. The second case is recognized by 'mount' keyword.

The first argument is a normal assign name. The following are the paths where assignment should be made. For example the first assign would be the same as:

```
Assign DPaint: Work:GFX/DPaint
```

and third is the same as:

```
Assign LIB: SC:LIB SC:LIB2 Development:store/LIB
```

Note, that you can specify assigns referring to another (unexisting) assigns. For example:

```
AAA:      BBB:
```


BBB: Work:

is correct.

When 'mount' keyword is given it indicates that a mount should be performed. The first argument is a device or a volume name. The second is the 'mount' keyword followed by arguments for 'Mount' command. The arguments are directly passed to the 'Mount' command - you can use multiple devices to mount. Also you can use FORM option to specify optional mountlist.

Important - when you specify automount by a volume name the corresponding device's mountentry MUST have its 'Mount' keyword set to 1 (i.e "immediate" mount). Why? Look at following example:

```
XH2:          mount XH2:
XFH_Works:    mount XH2:
```

When XFH_Works: is first referred to 'mount XH2:' will be executed. XH2: will be mounted but if it's "late" mount volume XFH_Works: won't appear! So you will get what you wouldn't want to... ('Please insert...').

Note, if either a path, an assign name or device name consists any blanks it must be enclosed in double-quotes (it's general rule in AmigaDos).

Summarizing - the template for

- assigns is NAME/A,TARGET/M/A
- mounts is NAME/A,MOUNT/S/A,DEVICE/M,FROM/K

VERY IMPORTANT: AutoAssign doesn't make *ANY* checks whether the assignsfile is correct! Any errors, like missing colon, missing quote, unexisting path won't be reported! Therefore you should edit the assignsfile carefully.

1.10 The Assign Wedge

The AssignWedge

As I have written above AutoAssign perfectly complements with such "Assign" commodities like AssignX, AssignWedge or ReqChange's AssignWedge. But to make it work AutoAssign *MUST* be run AFTER AssignWedge!!! If you put AutoAssign into WBStartup drawer STARTPRI tooltype should be set to lower value than STARTPRI value of AssignWedge! (If you are not familiar with STARTPRI tooltype refer to your Workbench Manual).

For your convenience AutoAssign's icon has STARTPRI tooltype set to -128 (the lowest value).

1.11 To Do

To Do

- o Installer script
- o A tool to extract all assigns from user-startup and create (or add to) the assignsfile (soon :-)
- o A hotkey to remove all previously made assigns (ie AutoAssign should be a commodity)
- o Polish and, maybe, French documentation.
- o AutoAssign prefs editor (?)

Note, that the last one depends on the feedback I will get from you.
(Just don't be lazy - write few words if you like & use AutoAssign)
For me it's convinient enough to edit assignsfile with an editor.

FEEDBACK REALLY ENCOURAGED!!! TELL ME YOUR IDEAS!!!

1.12 Author

Contacting the author

Any **ideas**, **bug reports**, **donations**, hardware (for example Cyberstorm turboboard for my A4000), etc. will be sincerely welcomed. Just contact me!

My addy:

Grzegorz Calkowski
Rzedowice
ul.Dobrodzienska 8
42-780 Dobrodzien
POLAND

Phone: +48 (0) 34-575875

E-mail: calkowsk@us.edu.pl

(Sorry, for my poor English...)

Always remember:

Beeting is beliving...

and Amiga is the best computer ever!

1.13 Acknowledgements

Acknowledgements

Thanks go to:

- o Michal "Dixan05" Prochownik for the game idea. Hope we'll realize the project :-)
- o Ryszard "Justus" Baginski for being "Marysia" ;)
- o Jacek "Shind" Grzegorzczuk for supporting me with additional 340MB harddisk and for loads of hot new stuff :-]
- o Pawel Wezgowiec for hard work with downloading Linux for me.
- o For develop team at C= (R.I.P) for the wonderful machine.
- o EScOm for giving us a ray of hope...

1.14 History

History

v1.00 (06.07.95)

- o First release.

v1.01 (07.07.95)

- o Doesn't lock the assignsfile for the whole run time. It allows for editing the assignsfile without quitting & re-entering AutoAssign. As a side-effect full path to the assignsfile must be given.

v1.10 (25.07.95)

- o Auto-mount feature added.
- o Support for multi-assigns added. Now uses ReadArgs() to parse.
- o Icon's STARTPRI tooltype set to -128 instead of -30. (To be safer :-)
- o Documentation reworked a bit.

v1.11 (27.07.95)

Oooopss... 1.10 was an *accidental* release - I forgot to include the latest compiled executables. (So, in fact, this is what 1.10 should be).

- o FROM keyword for mount didn't work. If specified _whole_ mount line was ignored! Fixed.

v1.12 (23.10.95)

- o Recompiled with some compiler's options changed - now ca. 500 shorter!
 - o Documentation update - included my E-Mail address! (finally, got it %-)
 - o Expect new _major_ update soon (-: When I finish my current -commercial-project :-)
-