

AGASSM

COLLABORATORS

	<i>TITLE :</i> AGASSM		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 20, 2024	

REVISION HISTORY

<i>NUMBER</i>	<i>DATE</i>	<i>DESCRIPTION</i>	<i>NAME</i>

Contents

1	AGASSM	1
1.1	main	1
1.2	help_about	2
1.3	help_new	2
1.4	help_windows	3
1.5	help_buttons	3
1.6	help_menus	6
1.7	help_macros	8
1.8	help_pack	8
1.9	help_sys	8
1.10	help_keystrokes	9
1.11	help_list	9
1.12	help_defaults	9
1.13	help_objects	9
1.14	help_ports	10
1.15	help_scripts	10
1.16	help_commands	11
1.17	help_acks	13
1.18	help_history	14
1.19	help_trouble	15
1.20	help_disc	16

Chapter 1

AGASSM

1.1 main

AGASSM v7.22 by Paul Vernon (15/1/96) AmigaGuide Help file.

- I. About
- II. Windows
- III. Buttons
- IV. Menus
- V. SlideShow List
- VI. ARexx and Commodity Ports
- VII. System Requirements for operation
- VIII. Acknowledgments & Contact Address
- IX. History
- X. TroubleShooting
- XI. Disclaimer

AGASSM v7.2x is Copyright (C) Paul Vernon 1996.

```

*****
*** IMPORTANT IMPORTANT IMPORTANT IMPORTANT IMPORTANT IMPORTANT ***
*** ***
*** AGASSM and associated files are copyrighted by Paul Vernon. ***
*** 5D Liceware and its appointed distributors have exclusive ***
*** distribution rights to the registered version. If you receive a ***
*** 'registered' version from elsewhere then please contact 5D so ***
*** that action can be taken. ***
*** ***
*****

```

1.2 help_about

AGASSM is designed to be an AGA chipset slideshow creator. AGASSM and its player both require OS3.0 or above to operate.

AGASSM is designed to be a quick system friendly SlideShow creation utility that takes advantage of OS 3.0 and above. The origins of AGASSM date back to early 1994 when I got my hands on a copy of Blitz Basic II. Thanks ACID for a great language! P.S. (Where's the support gone?)

AGASSM has a few tricks up its sleeve, so that you can produce some rather striking slideshows while still having all the advantages of OS 3.0 such as ARexx. AGASSM is able to display IFF-pictures (not 24-bit), play IFF-animations (Anim5), music modules (MED (MMD0 format) and ST), IFF-samples, and execute ARexx commands and Scripts as standard.

The useful aspect of AGASSM, is its **ARexx port**, which enables you to do things most Slideshow programs can't. That is utilise other peoples viewers and players to display or play file formats that AGASSM cannot normally cope with. For instance, AGASSM can cope with MED-Modules up to MMD0 but cannot play 8-track MED-Modules as standard. This however is not a problem as Teijo Kinnunens 'OctaMEDPlayer' has an ARexx port so, you can launch the player and load and play an 8 channel module all from AGASSM.

AGASSM was written entirely by myself (Paul Vernon) as a personal utility for my own use but I was convinced by my friends at University to publish it as Licenceware as they thought it looked 'dead professional!' well professional it isn't but I hope it goes some way to making the creation of slideshows a little bit easier.

I would appreciate it if when you copy the program for your friends, you leave out the registration file so that the save feature is disabled. If you don't, and a copy of your program gets back to me I shall withdraw my support to you by refusing to supply you with future versions! YOU HAVE BEEN WARNED!

AGASSP is freely distributable. AGASSM is most definitely NOT!

[What's new in version 7.22](#)

1.3 help_new

Here's a quick rundown of the additions and changes to AGASSM since the last version of AGASSM (6.4).

- Re-designed interface (altered/added windows)
- Filetype recognition system
- New fader routines
- Better implementation of user defined macros
- Now requires ARexx to operate
- Ability to alter sample playback rate
- Standardised file format for slideshows
- Full random access list editing (Add, Up, Down, Remove) (v7.21)
- Several Wipes now added. (v7.22)
- Transparent support for **PowerPacked** files
- Several other major/minor bug fixes.

For the first time ever AGASSM does not require the AGA chipset any more to operate and if you attempt to display a picture that requires the AGA chipset, you will be informed if your machine is incapable. AGASSP also incorporates this feature, and ignores AGA specific pictures on ECS machines! This also applies to AGA machines that do not have the SetPatch installed in their startup sequence. For these machines, the AGA screen modes are inaccessible until the SetPatch command is run.

On first looking at AGASSM 7.2x there is an instant difference with v6.4. The interface has been redesigned. I've been wanting to do it for a while but it seemed like a lot of trouble for a marginally better system. Anyway with the summer holiday upon me (and no job) I had nothing better to do so I redesigned the user interface. Hopefully, you will be able to see the advantages it brings.

Windows are smaller with the gadgets being more compact and following a much more logical layout with gadgets being grouped together according to their functions.

Due to the interface changes, I had to change the file-format for the slideshows slightly, so AGASSM 7.2 can't load in AGASSM 6.4 slideshows. Hopefully, this file format will be the standard for a while and I have given it a name! Its called 'SSM0' and its identifier can be found in the first four bytes of the slideshow file. The format is smaller than previous versions of AGASSM and contains more information!

The fade routine has been changed too. From 3 procedures and well over 60 lines of code to 2 procedures and about 8 lines of code. The speed increase is extremely noticeable on 256 colour pictures. The fade is also much much smoother.

The sample playback routine has also been improved, with the user now able to define the frequency of playback through the Object Information window, although the maximum sample frequency of a sample can only be 32767 Hz. This however should be sufficient for most samples.

As an extra, I've now incorporated a filetype recognition system that automatically identifies the type of object that you are attempting to load. Along with the filetypes AGASSM can load, I've also added some filetypes that it can't load so that AGASSM automatically rejects an object that it cannot handle.

The file-recognition system also works via the ARexx menu, so you can use the **ADDOBJECT** command with a wildcard and the objects will only be loaded if AGASSM can handle them! It also automatically configures the object with its specific object settings! Be warned however, it sometimes gets things wrong! Due to the spurious/random data in a file.

With reference to samples, the file-recognition system also looks inside sample files and finds the playback rate at which they were recorded. This is then used as the default, but can be changed by the user to play back at any rate.

To cope with all these changes, AGASSP has been updated to version 2.1, with a new look interface, incorporating a message box and an Error message box in its window. AGASSP is also now able to cope with multiple selection in Workbench and will happily play 1 slideshow directly after another and another and another and...

Also AGASSP incorporates the new fader routines and also supports the new file-format. (If it is really necessary, then I will code a conversion program to convert old slideshows (Tell me if you want one!))

1.4 help_windows

AGASSM is a modularised program with 4 separate windows with 4 distinct **functions**. The first window is the main slideshow creation window. The other 3 are support systems for this main window.

The main two windows are the initial window that is presented on booting AGASSM and the Object Information window. This can be accessed through the menus or by double-clicking on an object in the slideshow list.

Through the Object Information window, you are able to define several aspects of the object, telling AGASSM exactly what the object is and how it is to be dealt with. A few options in this window have been incorporated from the 'Extras' **menu** of version 6.4 of AGASSM and the 'Extras' menu has now been removed.

The remaining windows are the Defaults window to configure AGASSM for a slideshow and the ARexx Command Window. Information in the Defaults window can be saved to a config file, but it is not necessary as the information is also saved within each slideshow file. The ARexx Execute window enables you to execute an ARexx command line.

1.5 help_buttons

The Main Window

The main window is where most of the slideshow creation is done. Objects can be loaded into and removed from the slideshow and general aspects of the slideshow can be edited.

New Slideshow!

Clears the **SlideShow List** ready to create a new slideshow and resets all **defaults** to their last saved settings.

Play SlideShow

Plays the currently loaded slideshow that is in the list.

Load SlideShow

Brings up a requester to load a pre-made slideshow for editing. Slideshows are stored in an encrypted format and when viewed with any other system appear garbled.

Save SlideShow

Brings up a requester to save a slideshow for use with the player utility and for re-editing in the future.

Add

Enables you to load an **Object** into the current slideshow list after the currently selected object.

Up

Allows you to move an object in the list up.

Down

Allows you to move an object down the list.

Remove

Removes the currently selected Object from the SlideShow list.

Time :

You can use this gadget to determine how long AGASSM waits after displaying a picture before it displays/plays/executes the next Object in the list.

Loop :

Determines how long the slideshow is repeated. Be careful as if you leave it at its default setting, the slideshow will repeat infinitely. If you play the SlideShow by mistake and it begins to repeat, then hold down both mouse buttons until AGASSM returns you to the WorkBench.

Quit :

This gadget quits AGASSM.

The Object Information Window

The Object Information window is where you can edit several different aspects of the currently selected objects settings.

Object Type Cycle Gadget

This Cycle Gadget defines what the selected **Object** actually is.

Viewmode Cycle gadget

If the Object is a picture then it can be Normal, Extra Half Bright (Brite for all you silly people who spell it like that) or HAM. This cycle gadget selects the viewmode for each picture.

Module Type Cycle Gadget

If the current Object is a music **Module** then it can be either a MED or an SoundTracker module.

Stop Module

If a music module is playing then this gadget stops it playing.

View/Play Object

Displays, plays or executes the currently selected Object which can be identified by the text gadget to the left of this gadget.

Frames per (s) : (Frames per second)

Display the replay rate of the currently selected animation in the list.

Loop :

This can be set to loop the currently selected animation for up to 100 times.

Frequency

This displays the playback frequency of an IFF-8SVX sound sample.

You are able to set the playback frequency via the slider control beneath the information box.

Effects :

This gadget cycles through all the different options that can be performed on an object.

BackGnd

Makes the currently selected picture into a background that is displayed in between the other pictures in a slideshow.

Wait...

If a slideshow becomes too large to fit on one disk, then you can use this facility to carry on over several disks. For instance, you have an animation saying please insert disk 2. Select this object in the slideshow list, then select this option. After that, begin building the slideshow from your next disk and when you come to play the slideshow, the animation will repeat until the next disk is inserted! If however, you have a multi-drive system and the disk is already in a drive, the 'Please insert...' animation will not be displayed.

Fade

This facility is only available of pictures of the Amiga's Normal Viewmode. That is pictures of 2 to 256 colours. When this option is selected, the picture fades in from and out to black.

Loop Sample

This plays a looped IFF sample according to its internal settings.

Slide

This slides a picture up the screen.

HorizB

This draws the picture on the screen in a column by column fashion.

VertB

This draws the picture on the screen in a row by row fashion.

SpiralB

This draws the picture from the centre, spiralling outwards.

RandomB

This draws the picture using randomly placed blocks.

Blitter

This fires blocks onto the screen from the bottom left corner of the screen.

WARNING :- The blocks effects require standard sized screens or Video overscan screens. They cannot cope with wierd screen sizes. If you give AGASSM a wierd screen size, it may (or may not) crash!

To skip a picture that is being drawn with one of the effects, hold down the right mouse button.

Close

Closes the Object Information window.

The ARexx Command Line Window

With this window, you can directly execute or add a set of ARexx commands to a Slideshow.

Execute

Executes the current **ARexx** string in the String gadget. The string must be enclosed in speech marks i.e. "" with filenames being enclosed in apostrophes i.e. "

Add to SlideShow

Rather than executing the Command line, you can add it directly to a Slideshow.

Save as macro

Enables you to save the current argument in the requester as a macro for use within AGASSM.

Cancel

Closes the window without any effect.

A set of command's in the ARexx Execute Command Window must be surrounded by a pair of speech marks with filenames being surrounded by the apostrophe e.g.

```
"LOADSHOW 'WORK:GFX/MYSHOW.SHOW';EXIT 'Slideshow loaded';"
```

The semi-colon separates separate commands from each other, while the exit command must always be the last command in a command sequence. A message may be given by the EXIT argument but it is not necessary e.g.

```
"LOADSHOW 'WORK:GFX/MYSHOW.SHOW';EXIT;"
```

This does exactly the same as the previous command sequence with the exception that AGASSM does not display a message on completion of the command sequence.

The Defaults Window

This window enables you to set the default settings for each time you run AGASSM.

Audio Filter

Changes the status of the Audio Filter

System Requesters

Toggles the system requesters on and off (Useful for multi-disk slideshows)

Sample Buffer

Sets the size of the IFF sample diskplay routine. Around 16k (16384 bytes) should be enough for most purposes.

Hotkey

Allows the user to define AGASSM's commodity **HOTKEY**

(default :- ctrl lalt a)

Use

Implements the default settings.

Save

Saves and implements the default settings.

Cancel

Resets the defaults to their previous settings and closes the window with no effect.

1.6 help_menus

Most of the power of AGASSM is linked to its **SlideShow list** and the menu bar this gives a run down of the features of the menu bar.

Project

Load SlideShow A L

Bring up the file requester so that a slideshow can be loaded.

Save Slideshow A S

Save the current slideshow to a file.

Play Slideshow

Plays the current slideshow. To end a slideshow prematurely, hold down both mousebuttons until AGASSM or AGASSP returns you to the WorkBench.

New SlideShow

Clears the **SlideShow List** ready to create a new slideshow and resets all **defaults** to their last saved settings.

Defaults... A D

Enables the setting of various **defaults** for use with AGASSM

About... A ?

The About requester!

Hide H H

Quit A Q

Quits AGASSM!

Objects

Load Object A O

Loads an Object into the SlideShow list.

Object Type »

Defines what the **Object** actually is.

This menu option has several sub-options, these are :-

- Picture A P
- Animation A A
- Module A M
- ARexx-Script A R
- IFF-Sample A F

View Object A V

Displays, plays or executes the currently selected object.

Remove

Removes the last object in the Slideshow list.

Stop Audio

If a music module is playing this gadget stops it.

Information... A I

Opens the Object Information Window.

ARexx

Execute ARexx Script...

Brings up a file requester so that you might choose an ARexx script to execute.

Execute ARexx Command... A E

Brings up a text requester so that you can enter your own ARexx commands in a similar fashion the WorkBench's Execute Command function. Although, you can add the command directly into an AGASSM slideshow rather than execute it using the 'Add to Slideshow' Gadget in the requester.

Re-Read ARexx Macros...

The **ARexx** menu has a special feature, in that it assumes that any file it finds with the '.agassm' extension on the end in the 'REXX:' directory is an ARexx script and adds it to the menu. This way, AGASSM allows up to 150 self defined **macros** using the ARexx language as its core.

The Re-Read ARexx Macros option enables you to add/change/remove macros in the 'REXX:' directory and Re-Read the directory therefore updating the list of macros in the menus without Quitting and re-running AGASSM.

The user defined macros can also have their own user defined shortcuts in exactly the same way as normal menu items. You must be careful to give these macros unique shortcuts or clashes will occur. The process of giving the macro a keyboard shortcut is quite simple requiring you to be able to rename a file!

For example :-

To give the file REXX:Add-Directory.agassm a shortcut letter of 'C' the file must be renamed to 'REXX:Add-Directory,C.agassm' where the ',' separates the text appearing in the menu from the keyboard shortcut.

1.7 help_macros

Adding macros to the ARexx **menu** could not be easier. All you have to do to add a macro to AGASSM's menu is add the '.agassm' suffix to the filename and copy the macro into the 'REXX:' directory. The 'REXX:' directory is usually assigned to the 'S:' directory.

Up to 150 user defined macros can be added to AGASSM to make the creation of slideshows more simple. For instance one of the supplied macros will search through any directory that you specify and add any files that it recognises in that directory. This makes the creation of slideshows easy as once all your pictures/animations/music files are placed in their directories on your slideshow disk all you have to do is 'Add' each directory to the slideshow!

Macros can be created using your favourite text editor and once they are in the 'REXX:' directory, they can be recognised by their '.agassm' suffix and loaded into the menu using **Re-Read ARexx Macros**. Simple!

1.8 help_pack

AGASSM now supports crunched files using the PowerPacker.library. This will allow you to fit more onto one disk.

All objects except ARexx scripts can be used in a crunched form. That is to say, you can use crunched IFF Pictures/Animations and Samples. You can also use crunched music modules of both MMD0 and ST format.

There is a slight drawback to using Power Packed files however. AGASSM's file recognition system cannot look inside the file to identify what type of object it is, so you have to tell it. Once you have done that, AGASSM behaves as if the file were a normal un-crunched file.

AGASSM requires large amounts of memory for the Power Packer feature to work well (4 Mb at least I should say). If you are running a standard A1200 like me, then you will be able to use Power Packer on the small pictures. Forget about using a 640 by 512, 256 colour picture with an effect because power packer uses internal memory buffers as well as the effect, and then there's the screen to open!!

1.9 help_sys

Operating System :- 2.1 or above

Memory Requirements :- 2Mb (4Mb is better)

Disk Drives required :- 1 Floppy Drive (2 drives are better)

(Hard Drive is recommended)

Software Libraries needed :- PowerPacker.library

RexxSysLib.library

Mathieedoubbas.library

Commodities.library *

* AGASSM only

AGASSM and AGASSP have been tested on the following machines :-

A600 '000 7Mhz 2Mb (Does not work as yet HMMMM.... :-())

A1200 '020 14Mhz 2Mb

A1200 '020 14Mhz 2Mb + 4Mb

A1200 '030 50Mhz 2Mb + 4Mb

A1200 '030 50Mhz 2Mb + 4Mb + 12Mb (Virtual Memory using VMM)

1.10 help_keystrokes

AGASSM has several keyboard short-cuts that are designed to speed up the use of the program, but as this program is entirely mouse driven, I find that they slow me down. The keyboard short-cuts are available through the [menu](#) section.

Your own keyboard shortcuts can be added to AGASSM's ARexx menu for your user defined macros. The process to do so is located in the ARexx section of the [menus](#) guide.

1.11 help_list

The Slideshow list is the heart of AGASSM's ease of use and power. It enables you to add objects of many [formats](#) to a slideshow so that it can be a full Multimedia experience. The List enables you to build the general structure of the slideshow, then using AGASSM's other features, you can develop the general structure and add little details to the slideshow that make it look that little bit better. The current objects name is displayed in the Text requester below the list and can be changed via this Text requester.

1.12 help_defaults

AGASSM has several default settings that can be changed from within the Defaults window. These default settings are :-

The Audio Filter (On or Off)

The System Requesters (On or Off) (Useful for multi-disk slideshows)

The Sample disk buffer size for samples over 128k in length.

The [Commodity](#) HotKey

1.13 help_objects

AGASSM is not just a picture display utility. It can do other things too such as play music modules, samples and animations. It can even execute ARexx scripts from within a slideshow. For this reason, you need to define what each object is in the slideshow.

An object can be of 5 different types. These are :-

- IFF Picture
 - IFF Animation (Anim 5)
 - Music Module (MED or ST format)
 - IFF Samples
-

· ARexx Scripts

Through the use of ARexx scripts, this range of object types can be extended by the remote use of other programs capabilities.

IFF-Picture A P

This can be any standard IFF-Picture that can be created from within for instance DeLuxe Paint IV AGA ®.

IFF-Animations A A

These must be of the Anim 5 format and can be in any viewmode (Normal/Half Bright/HAM)

Music Modules A M

The MED module must be of the MMD0 format as AGASSM does not directly support MED modules of higher versions.

The ST module player plays most any ST-module, but I have experienced trouble with very old ST-modules.

IFF samples A [

Samples must be mono, but can be of any length. If they are over 128K however, AGASSM plays them directly from disk and looping is quirky with these long samples.

ARexx Scripts A R

ARexx scripts can be included to either trigger outside programs or affect the way in which AGASSM itself is running.

1.14 help_ports

AGASSM is equipped with both an ARexx and a Commodity port. Indeed AGASSM will refuse to run if ARexx has not been initiated on your system. To initiate ARexx, all you have to do is run the REXXMast® program in the System drawer of your System Disk (WorkBench).

ARexx Port

The ARexx port has along with it several Built-in **commands** that can be used to control AGASSM from remote applications. AGASSM can also control remote applications using their in-built commands. For instance, AGASSM has no in-built support for playing OctaMED® songs or modules, but can still be used to play them via an ARexx script controlling the OctaMED player.

As well as ARexx **Scripts**, AGASSM also provides support for the direct executing of ARexx commands via the ARexx menu.

Commodity Port

The Commodity port is a simple affair, letting you open and close the main AGASSM display while keeping AGASSM running in the background. This really is just an advanced sleep facility. You can also quit AGASSM using the CX utility.

1.15 help_scripts

ARexx scripts can be either executed from the **menu** or from within an AGASSM slideshow. They can be used to either control AGASSM or some external program that also supports ARexx.

Here are a few example scripts for use within AGASSM.

OctaMED player ® routine

```
/* OctaMED Player DEMO */
```

```
ADDRESS COMMAND 'C:Run Work:Music/OctaMED/Utilities/OctaMEDPlayer'
```

```
ADDRESS COMMAND 'Wait 2'
```

```
ADDRESS 'OCTAMEDPLAYER'
```

```
LoadMod 'Work:Music/OctaMED/Mods/FairLigt.mod'
```

```
Play
```

```

Exit 'OctaMed has started playing' /* This message is displayed in AGASSM */
clariSSA @ demo script
/* AGASSM ARexx Script to play a demo anim from clariSSA*/
exists = Show('P','CLARISSA')
If exists = 0 then do
EXIT 'Unable to find clariSSA'
End
Options RESULTS
Address clariSSA
Requeststatus OFF
LoadAnim MASTER Name 'ClariSSA:SSA-Anim/SSADemoAnim'
PlayAnim 1 Screens
CloseAnim MASTER
EXIT

```

1.16 help_commands

AGASSM contains some internal ARexx commands that are used to control AGASSM from a remote piece of software. This allows information to be exchanged and the creation of a slideshow to occur automatically! As I don't actually own an ARexx manual, Commodore only put it in the A4000 package, I think I've not done too bad with these commands!

As with the GUI of AGASSM, certain features of this interface only work with certain object types. It is pretty obvious which commands affect which object types.

ADDOBJECT [filename] <objecttype [viewmode]> *

Adds an object to the current Slideshow List. Wildcards can be used in the filename.

SETOBJECTTYPE <filename> <objecttype [viewmode]> *

Sets a specified object to a specific format. Wildcards can be used in the filename.

<objecttype> = 0 - Picture

1 - Animation

2 - Module

3 - ARexx Script

4 - IFF Sample

[viewmode] = 0 - Normal OR 0 - MED module

1 - Extra Half Bright 1 - ST module

2 - HAM6/8

VIEWOBJECT <filename> *

Displays, plays or executes the selected object. Wildcards can be used.

REMOVE <filename> *

Removes an object from the slideshow list. Wildcards can be used.

STOPAUDIO

If a music module has been launched from within AGASSM this will stop it.

ADDEFFECT <filename> <extra> [suboption] *

Adds an effect or 'extra' to an object.

<extra> = "WAIT|FADE|LOOPI|BACKGND|SLIDE|BLOCKS"

where the BLOCKS option has the sub-options.

"HORIZ|VERT|SPIRAL|RANDOM|BLITTER|BLINDS|L-R|U-D"

REMEFFECT <filename> *

Removes an effect or 'extra' from an object.

SETANIMSPEED <filename> <fps> *

<fps> = 1 to 25

SETANIMLOOP <filename> <number of loops> *

<number of loops> = 1 to 100

LOADSHOW [filename] *

If the filename is omitted, a file requester is opened.

SAVESHOW [filename] *

If the filename is omitted, a file requester is opened.

NEWSHOW *

Resets AGASSM and clears the slideshow list.

SETDISPTIME <filename> <time> *

<time> = 0 to 100 in steps of 0.1 if set to 0 then AGASSM will wait for the mouse to be pressed. Wildcards can be used.

SETSHOWLOOP <number of loops> *

<number of loops> = 0 to 100 in steps of 1 if set to 0 then the slideshow will loop infinitely.

REQUESTERS <ON|OFF>

Turn the system requesters on or off. This is useful for multidisk slideshows.

FILTER <ON|OFF>

Controls the audio filters status.

SAMPLEBUFFER <size>

Sets the size of the replay buffer for samples over 128k in length.

<size> = 1k to 64k in steps of one. Around 16k should be fine for most.

SAMPLERATE [filename] [rate] *

Sets the playback rate of the specific sample in Hz. Wildcards can be used. Rates are valid between 3000 and 32767 Hz.

SAMPLEVOLUME [volume]

Sets the volume at which samples are played. Valid values are in the range 0-63.

SHOWMSG [time] <message>

Displays a message in the information bar for a specified amount of time. The time can be between 1 and 5 seconds. The message can be upto 7 words long where a word is defined as anything with a <SPACE> character either side of it so...

ShowMsg 1 'Hello this is a message !'

would be a message of 6 words. Whereas ...

```
ShowMsg 1 'Hello this is a message!'
```

would be five words.

```
QUIT *
```

Closes AGASSM down.

* - These commands are not available in AGASSP but are available in AGASSM.

Using these AReXX Commands

To use these commands, you can either incorporate them into a **Script** or you can include them directly into a slideshow. Either way, you have the whole of the AReXX language at your disposal to use as a programming language for all the programs that use it. To add a set of AReXX commands to a slideshow, you must type them in the 'Execute AReXX Command' **window** and click on the 'Add to Slideshow' button. This then adds the command into the slideshow as the next function in the slideshow.

1.17 help_acks

Amiga, Workbench, A1200 and A4000 are trademarks of Escom (Amiga Technologies GmbH).

DeLuxe Paint is a trademark of Electronic Arts.

ClariSSA is a trademark of ProDAD.

OctaMED and OctaMEDPlayer are Copyright © Ray Burt-Frost (RBF Software) & Teijo Kinnunen.

PowerPacker and PowerPacker.library are Copyright © Nico François.

VMM is Copyright © Martin Apel.

Thanks to all who have used and tested this program. Especially Stephen who used to crash AGASSM with an uncanny regularity that was un-nerving. Anyway I fixed all the bugs that he found! Also to Phil Wilkinson of SaddleTramps PD and 5D Licenceware for giving me several ideas for additions to the program and for bug testing on OS 2.x machines (Still doesn't work :-). I couldn't have fixed or found all the bugs without his help!

If you write a particularly good macro for AGASSM and you think other users will benefit from it, then please send it to me and I will be pleased to add it to the distribution disk so that it will receive the widest possible audience. You will also be added to the acknowledgements in this file.

If you wish to register AGASSM or you have any bug reports or suggestions then you can contact me at my personal address.

To register send £2.95 (Sterling) with your postal address to :-

Cheques/P.O's payable to SaddleTramps PD.

5D Licenceware,

1 Lower Mill Close,

GoldThorpe,

Rotherham,

South Yorkshire,

S63 9BY.

Personal Address :- (BUG REPORTS and SUGGESTIONS only)

Paul Vernon

5 Dale Road

Dronfield

Sheffield

S18 6YG

ENGLAND.

Or, send an E-Mail to me at dbae260@uce.ac.uk. This E-Mail address is good until May 1996 when I finish my second year of my degree.

1.18 help_history

AGASSM was designed and created over several months with only my personal use in mind therefore, most of the features that I have added are ones that I myself find useful. Version numbers before v2.5 were incredibly basic and therefore I won't mention them.

v2.5 2/2/94 Very basic using pictures only and was OS1.x compatible

v4.1 8/3/94 Yes, lots of revisions including what I thought was OS2.x and above but turned out to be OS3.x and above.

Complete code re-write from scratch.

Pictures, Animations and modules are catered for.

Background pictures could be used (thanks to a suggestion)

Very Large interface. (Really needed its own screen!)

Only one module per slideshow.

Sleep facility.

v4.5 24/3/94 Cosmetic changes to GUI.

Lots of bug fixes.

Code was made modular for easier de-bugging and general tidiness.

v5.1 1/4/94 More GUI changes.

Multiple modules in a single slideshow.

Yet more modularisation of code bringing a 20K reduction in executable size!

v5.2 24/4/94 ARexx interface added. (Test version only)

v5.3 25/4/94 Bug fixes. Lots of them appeared after the ARexx module was added but were nothing to do with ARexx as such!

v5.4 14/6/94 Pull down menus added giving the ability of keyboard shortcuts.

v5.6 11/7/94 OS2.x requesters added (previously they were all my own code!)

Improved ARexx support (more commands!)

v5.9 29/7/94 IFF Sample support added.

Defaults editor window added.

More ARexx improvements.

v6.1 11/8/94 Keyfile protection added.

ARexx re-written with more commands also.

v6.2 4/12/94 Wildcard support added to ARexx interface

(Thanks for the autodocs info James)

v6.3 14/1/95 Added a commodity port. AGASSM can be made to quit and hide from the Commodities Exchange program.

v6.4 14/7/95 Minor bug fixes and a better event handling system.

v7.1 22/8/95 (Test purposes only)

Totally re-designed GUI.

More changes to ARexx.

OS2.x+ rather than OS3.x+

Compatible with all chipsets (OCS, ECS and AGA).

I finally fixed the event handling bug in AGASSM!

Full list editing features (Add, Up, Down and Remove)

v7.21 9/11/95 Extra effects.

Better memory management.

CPU sensitive, changing the speed of the effects.

Basically all the prototype **features of v7.22**

v7.22 10/1/96 Bug fixes so that AGASSM runs with Virtual Memory.

(I bought an accelerator with an MMU!)

Enforcer Hits removed and fixed!

PAL/NTSC mode switching removed due to Enforcer Hit.

1.19 help_trouble

Well, here it is! If you have any problems then I suppose the answer should be here. If you have a problem, and the answer is not here and you manage to solve it on your own then write it down and send it to me so that I can add it to this section or change the code to suit.

Q · AGASSM appears to load but asks for the 'RexxMast' program.

What is 'RexxMast' and where is it ?

A · 'RexxMast' is a type of programming language that enables utilities to share and exchange information and also remotely control other tasks that are running. It can be found in the System drawer of your Workbench disk. AGASSM requires the use of ARexx for some of its operations and therefore will refuse to run unless 'RexxMast' has been activated and **ARexx** is present on the system.

Q · I have an AGA machine but AGASSM says that only the ECS modes are available. Why is this?

A · When AGASSM is first run, it checks to see whether the Setpatch program has been run. If not, then the extra AGA features are not available to any OS friendly program. To enable AGA all you have to do is run the Setpatch command that is found in your 'C:' directory and re-run AGASSM. The about requester should now tell you that your current chipset is AGA.

Q · Although some slider gadgets are not disabled, they don't work!

A · This is because you have not selected an object at all or have not selected an object of the correct type!

Q · AGASSM tells me that I have too many items in a **list** when I try to add any more.

A · This is because AGASSM only allows 250 objects in a slideshow. Do you want more ? (500, 1000, ...)

Q · AGASSM says it has found an Unknown Error. What is it?

A · This is mainly for my testing purposes and you should never see this. If you do you may get a program reference name or a message to report this error to the author. In both cases I would ask that you inform me of all the details so that I can re-create the error so that I can eradicate it. Having said that I've not seen this error in ages but no program is 100% error free. Even this one.

Q · AGASSM is not using the effect when displaying one of the pictures in a slideshow, but AGASSP does. Why?

A · This is a border line low memory scenario where AGASSM occupies more memory than AGASSP and in 2Mb or less machines, this is enough for AGASSM to run out of memory to carry out the effect so it just shows the picture instead. In really low memory situations, AGASSM does not even open a screen.

1.20 help_disc

Neither Paul Vernon or 5D Licenceware and its appointed distributors guarantee the stability of this program.

They are not liable for any harm that comes to you, your system or the data held by your system through the use or misuse of AGASSM and associated programs.
