

Visage

Magnus Holmgren

Copyright © Copyright1995-97 by Magnus Holmgren

COLLABORATORS

	<i>TITLE :</i> Visage		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Magnus Holmgren	February 12, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Visage	1
1.1	Visage.guide	1
1.2	Visage.guide/Introduction	1
1.3	Visage.guide/Features	2
1.4	Visage.guide/Legal information	2
1.5	Visage.guide/Legal issues/License	3
1.6	Visage.guide/Legal issues/No warranty	3
1.7	Visage.guide/Requirements	3
1.8	Visage.guide/Requirements/Required libraries	3
1.9	Visage.guide/Arguments	4
1.10	Visage.guide/Arguments/VisageOpts	4
1.11	Visage.guide/Arguments/FILES	5
1.12	Visage.guide/Arguments/BORDERBLANK	6
1.13	Visage.guide/Arguments/COLOR	6
1.14	Visage.guide/Arguments/COMMAND	6
1.15	Visage.guide/Arguments/DITHER	7
1.16	Visage.guide/Arguments/DRAWER	7
1.17	Visage.guide/Arguments/MODE	8
1.18	Visage.guide/Arguments/MONITOR	8
1.19	Visage.guide/Arguments/OVERSCAN	8
1.20	Visage.guide/Arguments/PASSWORD	9
1.21	Visage.guide/Arguments/PATTERN	9
1.22	Visage.guide/Arguments/BUFSIZE	9
1.23	Visage.guide/Arguments/DELAY	9
1.24	Visage.guide/Arguments/FADE	10
1.25	Visage.guide/Arguments/GAMMA	10
1.26	Visage.guide/Arguments/TIMES	10
1.27	Visage.guide/Arguments/TOOLPRI	10
1.28	Visage.guide/Arguments/UNIT	11
1.29	Visage.guide/Arguments/ALL	11

1.30	Visage.guide/Arguments/BESTMODE	11
1.31	Visage.guide/Arguments/CENTER	11
1.32	Visage.guide/Arguments/CLIPBOARD	11
1.33	Visage.guide/Arguments/COMPACT	12
1.34	Visage.guide/Arguments/CYCLE	12
1.35	Visage.guide/Arguments/DATATYPES	12
1.36	Visage.guide/Arguments/ECS	12
1.37	Visage.guide/Arguments/FOREVER	12
1.38	Visage.guide/Arguments/GRAY	13
1.39	Visage.guide/Arguments/HAM	13
1.40	Visage.guide/Arguments/INFO	13
1.41	Visage.guide/Arguments/LATELOAD	13
1.42	Visage.guide/Arguments/LOWMEM	13
1.43	Visage.guide/Arguments/NOAUTOSCROLL	14
1.44	Visage.guide/Arguments/NOBUSY	14
1.45	Visage.guide/Arguments/NOCLICK	14
1.46	Visage.guide/Arguments/NODATATYPES	14
1.47	Visage.guide/Arguments/NOENV	14
1.48	Visage.guide/Arguments/NOFILECHECK	14
1.49	Visage.guide/Arguments/NOFLICKER	15
1.50	Visage.guide/Arguments/NOFOREIGN	15
1.51	Visage.guide/Arguments/NOGIF	15
1.52	Visage.guide/Arguments/NOICON	15
1.53	Visage.guide/Arguments/NOIFF	15
1.54	Visage.guide/Arguments/NOJPEG	16
1.55	Visage.guide/Arguments/NOPNG	16
1.56	Visage.guide/Arguments/NORMALNAMES	16
1.57	Visage.guide/Arguments/NOWARN	16
1.58	Visage.guide/Arguments/POINTER	16
1.59	Visage.guide/Arguments/QUIET	16
1.60	Visage.guide/Arguments/RANDOM	17
1.61	Visage.guide/Arguments/RTG	17
1.62	Visage.guide/Arguments/SCALE	17
1.63	Visage.guide/Arguments/SHOWINFO	17
1.64	Visage.guide/Arguments/SORT	18
1.65	Visage.guide/Arguments/TEST	18
1.66	Visage.guide/Arguments/TIME	18
1.67	Visage.guide/Arguments/TOFRONT	19
1.68	Visage.guide/Arguments/VERBOSE	19

1.69	Visage.guide/Arguments/VCENTER	19
1.70	Visage.guide/Arguments/WAITFORPIC	19
1.71	Visage.guide/Arguments/WBMONITOR	20
1.72	Visage.guide/Viewing keys	20
1.73	Visage.guide/Algorithmic mode names	20
1.74	Visage.guide/Algorithmic mode names/Mode name parts	21
1.75	Visage.guide/Algorithmic mode names/Examples	21
1.76	Visage.guide/Algorithmic mode names/Notes	21
1.77	Visage.guide/Rendered pictures	22
1.78	Visage.guide/Included programs	22
1.79	Visage.guide/Included programs/DeleteLink	22
1.80	Visage.guide/Included programs/GetModeID	23
1.81	Visage.guide/Included programs/Kill	23
1.82	Visage.guide/Included programs/MakeLink	24
1.83	Visage.guide/Included programs/UnpackILBM	25
1.84	Visage.guide/Included programs/UpdateTT	25
1.85	Visage.guide/Notes	26
1.86	Visage.guide/Notes/Back picture visible	26
1.87	Visage.guide/Notes/Bad aspect	26
1.88	Visage.guide/Notes/Graphic cards	26
1.89	Visage.guide/Notes/Interlaced PNG and JPEG pictures	27
1.90	Visage.guide/Notes/Mouse jumping	27
1.91	Visage.guide/Notes/Pipe reading	27
1.92	Visage.guide/Notes/PNG support	28
1.93	Visage.guide/Notes/Screen centering	28
1.94	Visage.guide/Notes/Tooltypes ignored	28
1.95	Visage.guide/The future	28
1.96	Visage.guide/Acknowledgements	29
1.97	Visage.guide/Author information	29
1.98	Visage.guide/Program history	30
1.99	Visage.Guide/Program history/Version 39.0	30
1.100	Visage.Guide/Program history/Version 39.1	30
1.101	Visage.Guide/Program history/Version 39.2	31
1.102	Visage.Guide/Program history/Version 39.3	33
1.103	Visage.Guide/Program history/Version 39.4	33
1.104	Visage.Guide/Program history/Version 39.5	34
1.105	Visage.Guide/Program history/Version 39.6	34
1.106	Visage.Guide/Program history/Version 39.7	35
1.107	Visage.Guide/Program history/Version 39.8	35

1.108 Visage.Guide/Program history/Version 39.9	35
1.109 Visage.Guide/Program history/Version 39.10	36
1.110 Visage.Guide/Program history/Version 39.11	36
1.111 Visage.Guide/Program history/Version 39.12	37
1.112 Visage.guide/Index	38

Chapter 1

Visage

1.1 Visage.guide

Visage 39.12

Copyright © 1995-97 by Magnus Holmgren

Release date: 97-04-27

Welcome to Visage, a multi format picture viewer for the AmigaOS (version 3.0 or higher).

Introduction Why yet another picture viewer? **Features** What Visage can do. **Legal information** Distribution conditions, etc. **Requirements** What Visage needs to run.

Usage All arguments described. **Viewing keys** Keys you can use while viewing pictures. **Mode names** How to specify screen mode names. **Rendered pictures** What is a rendered picture ? **Included programs** Some utilities that are included.

Notes Some (sometimes odd) behaviour explained. **The future** What will happen with Visage? **Acknowledgements** Who did what? **Author** Address to the author of Visage. **Program history** What has changed?

Index Chapter and topic index.

Note: This manual needs AmigaGuide version 40 or better for correct display. The manual should be readable with earlier versions, but it will not look that good.

1.2 Visage.guide/Introduction

Introduction

<Sigh> Yet another picture viewer. :)

Oh well, what can I say? There was no picture viewer available that suited all my needs. They were lacking such trivial things as looking for external break signals, and more important things such as good monitor support.

But Visage is different. It has many features; some are not available in any other picture viewer for the Amiga (as far as I know). Since some of these features rely on things introduced in OS 3.0, this program requires OS 3.0. There are several viewers that work on OS 2.0, so it is not a major problem, IMHO.

The goal was a rather fast viewer, with many features. Picture quality was not the most important thing (this mainly applies to the HAM and color rendering).

See also: [Features](#) [Legal information](#) [Requirements](#) [Rendered pictures](#)

1.3 Visage.guide/Features

Features

Displays IFF ILBM, JPEG, PNG and datatype pictures.

The internal viewers (IFF ILBM, PNG and JPEG) use asynchronous IO for high performance. The ILBM decompressor is written in optimized assembler for speed.

Full AA, ECS and RTG (I hope! ;) support.

Supports the IFF ILBM chunks CLUT (Color LookUp Table; only RGB), PCHG (Palette CHanGe) and SHAM (Sliced HAM).

Supports CyberGraphX and picture.datatype version 43.

Displays "deep" pictures in all formats (including datatypes, if needed) on ECS Amigas, rendered in HAM, "normal" color (graphics card recommended), high- or truecolor (CyberGraphX and graphics card needed) or grayscale.

IFF ILBM pictures can be Xpk compressed.

Fast HAM rendering of pictures, using code written by Rafael D'Halleweyn.

Can **scale** any **rendered** picture to fit the screen.

Extensive monitor support. Uses a BestModeID()-like function to find a suitable mode (if needed or requested).

Any native Amiga screen **mode** can be specified with **text** ! No numbers needed. But you can use numbers, if you wish. Or a requester.

Simple promotion of pictures to any installed **monitor** . No hardcoded command line options for this. Even a requester can be used.

The internal viewers can read their data from a **named pipe** .

Several slideshow options, including a **random** and a **sort** mode, making Visage ideal for use with screen blankers as an external module.

Background loading of the next picture while viewing the current, if there is enough memory (this can be disabled).

When using background loading, the new pictures do not open in front of all other screens (unless told to). The two screens are "attached" to each other.

Can **fade** pictures (but not HAM, SHAM or PCHG pictures, nor pictures displayed in high- or truecolor modes).

Default options can be placed in an **environment variable** .

Invisible screen arranging gadgets.

Can execute a **command** for a picture.

Can show **information** about the pictures, **test** them or **time** them.

Horizontal and **vertical** screen centering.

Can be made resident.

System friendly color **cycling** (no interrupt).

Several other small features not mentioned here. :)

It is freeware! No need to pay anything (but that does not mean **I** would not appreciate a donation of any kind! :).

A couple of small and useful utilities **included** .

1.4 Visage.guide/Legal information

Legal information

License Distribution conditions. **No warranty** Standard disclaimer.

This software is based in part on the work of the Independent JPEG Group.

1.5 Visage.guide/Legal issues/License

License

Visage is released under the concept of freeware. This means you are allowed to use and copy this program freely, as long as the following requirements are fulfilled:

All files are copied without any alterations or modifications. If any extra files are added, it must be obvious that they do not belong to the original distribution, and that they do not need to be included in any redistribution. Exception: So called "BBS ads" may not be added.

The copying is done on a noncommercial basis. A small fee to cover media costs etc. may be charged.

The copier is not claiming the copyright of this program.

Any exceptions from the above require a written permission from the author.

If you want to publish this program on a cover disk or similar, contact me first for approval (to make sure you have the latest version etc). I then expect a copy of the issue of the magazine in return (additional contributions are welcomed :).

1.6 Visage.guide/Legal issues/No warranty

No warranty

There is no warranty for the programs, to the extent permitted by applicable law. Except when otherwise stated in writing the copyright holder and/or other parties provide the programs "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the programs is with you. Should the programs prove defective, you assume the cost of all necessary servicing, repair or correction.

In no event unless required by applicable law or agreed to in writing will any copyright holder, or any other party who may redistribute the programs as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the programs (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the programs to operate with any other programs), even if such holder or other party has been advised of the possibility of such damages.

1.7 Visage.guide/Requirements

Requirements

Visage should work on any Amiga that has OS 3.0 or higher. 512 KB of memory is enough for operation, but more memory is indeed recommended (especially if you want to view large pictures). There are no required disk based libraries, although Visage will be limited if some are not available.

I decided to let Visage require OS 3.0, since I otherwise would need to rewrite several new functions in OS 3.0, or write code to work around bugs. One feature cannot be implemented at all using OS 2.0. Despite all this, I might make Visage support OS 2.0 in the future (if I get requests for it. Have not got any yet... ;).

See also: [Required libraries](#)

1.8 Visage.guide/Requirements/Required libraries

Required libraries

The following disk based libraries are not necessary for Visage to operate, but certain things will not work if they are not available: asl.library version 38 or higher For the file, screen mode and monitor (!) requesters. If not available, you will get a warning if you try to use these requesters.

cybergraphics.library version 40 or higher For rendering to screens deeper than eight bits.

datatypes.library version 39 or higher For the datatypes reader. To work properly, suitable datatypes are needed as well. Some datatypes are included with the system; several exist as different kinds of freely distributable software.

iffparse.library version 39 or higher For the IFF ILBM reader.

mathieeedoubbas.library version 37 or higher, and mathieeedoubtrans.library version 37 or higher For the PNG reader (the gamma correction).

reqtools.library version 38 or higher For the password requester that appears if an Xpk-encrypted picture is encountered, but no password was specified. ReqTools can be found on Aminet and BBS:es.

xpkmaster.library To view Xpk compressed or encrypted IFF ILBM pictures. For it to work properly, suitable sublibraries are needed as well. Xpk can be found on Aminet and BBS:es. For best operation, XPK version 3 or higher is recommended.

1.9 Visage.guide/Arguments

Arguments

Visage can be started from a Shell or the Workbench. For Shell users it should be noted that Visage is pure, and may be made resident.

In the case of Workbench start, the tooltypes may be specified in either the program icon, the picture icon(s), or any other icon (preferably without an associated file). All tooltypes are collected (tooltypes found later on overrides earlier ones) and parsed before viewing the pictures.

VisageOpts can be used to specify new defaults.

For both Workbench and Shell usage, the following arguments/tooltypes can be specified:

Text arguments **FILES** Files to view. **BORDERBLANK** Set border blank mode. **COLOR** Render mode for deep pictures. **COMMAND** Command to execute, if requested. **DITHER** Dither type to use, when needed. **DRAWER** Initial drawer for file requester. **MODE** Screen mode to use. **MONITOR** Monitor to use. **OVERSCAN** Overscan type to use. **PASSWORD** Password for Xpk-encrypted files. **PATTERN** Initial pattern for file requester.

Numeric arguments **BUFSIZE** IO buffer size. **DELAY** Time to wait between pictures. **FADE** Speed for picture fade. **GAMMA** Gamma of your screen. **TIMES** Number of times to display the pictures. **TOOLPRI** Task priority for Visage. **UNIT** Clipboard unit to read from.

Switches **ALL** Enter all drawers encountered. **BESTMODE** Less strict mode promotion. **CENTER** Center the pictures. **CLIPBOARD** Read from the clipboard. **COMPACT** Use a more compact VERBOSE output. **CYCLE** Start color cycling automatically. **DATATYPES** Try to use a datatype. **ECS** Use an ECS-compatible screen depth. **FOREVER** View all files until aborted. **GRAY** Render pictures in grayscale. **HAM** Use HAM "when in doubt". **INFO** Only show information about pictures. **LATELOAD** Read next picture after timeout. **LOWMEM** No background reading. **NOAUTOSCROLL** Disable autoscrolling. **NOBUSY** Do not show any busy pointer. **NOCLICK** Do not listen to mouse clicks. **NODATATYPES** Do not use datatypes. **NOENV** Do not read options in VisageOpts. **NOFILECHECK** Do no extra check for file existence. **NOFLICKER** Do not use laced screen modes. **NOFOREIGN** Ignore certain screen modes. **NOGIF** Use datatype for GIF pictures. **NOICON** Ignore any '.info' files found. **NOIFF** Use datatype for IFF pictures. **NOJPEG** Use datatype for JPEG pictures. **NOPNG** Use datatype for PNG pictures. **NORMALNAMES** Use normal display mode names. **NOWARN** Disable warning messages. **POINTER** Do not blank the mouse pointer. **QUIET** Disable most output. **RANDOM** View pictures in random order. **RTG** Improve behaviour with some graphic cards. **SCALE** Scale rendered pictures to fit screen. **SHOWINFO** Display picture information on picture. **SORT** View pictures in alphabetical order. **TEST** Test pictures, do not display. **TIME** Display loading time. **TOFRONT** Bring newly loaded pictures to the front. **VCENTER** Center pictures vertically. **VERBOSE** Display extra information. **WAITFORPIC** Wait for next picture to load. **WBMONITOR** Use same monitor as Workbench.

(61 arguments! ;)

1.10 Visage.guide/Arguments/VisageOpts

VisageOpts

New default values may be specified by entering them in the VisageOpts configuration file or environment variable. The search order is as follows:

If the environment variable ConfigPath¹ exist, then try to read the file VisageOpts in the drawer specified by ConfigPath.

If the environment variable or the file in the ConfigPath drawer does not exist, then read the environment variable VisageOpts, if present.

Put all arguments on any number of lines, similar to if Visage was started from the Shell. Comments start with a '#' or ';' at the start of the line, and they end at the end of that same line.

To disable the reading of VisageOpts (e.g. in a script), use the NOENV switch. This disables both the environment variable and the ConfigPath file.

If a switch is specified in VisageOpts, then by specifying it again on the command line you will turn off that switch. Thus, if CENTER is found in VisageOpts, and CENTER is specified on the command line (or in the tooltypes), no centering will occur.

Text or numeric arguments specified on the command line will simply override any arguments specified in VisageOpts. Most text arguments interprets the empty string ("") as if the string was not specified (i.e., it reverts to the default value).

Two arguments in VisageOpts will always be ignored: FILES and NOENV. You will not get a warning or anything if they are specified; they are silently ignored. This has the side effect that unknown options are ignored (they are "swallowed" by the FILES argument).

See also: **FILES CENTER NOENV**

-- Footnotes --

¹ If you specify 'Progdir:' here (to get the same drawer as where the program resides), please remember that you should not make the program resident in that case. If you do, you will get 'Please insert volume Progdir:' requesters, which can be annoying.

The name of the ConfigPath variable was deliberately chosen to be general. The hope is that other programs also will use this, at least as a part of their configuration file search path.

1.11 Visage.guide/Arguments/FILES

Files

Here you specify the files you wish to view. You may enter any number of files and drawers here, using patterns if you like.

If you enter the name of a drawer, or no files at all, then a file requester will open, located in that drawer (unless ALL has been specified). If you in this requester specify the name of a file that does not exist, then Visage will treat it as a pattern. However, if the NOFILECHECK option have been specified, Visage will always treat the specified name as a pattern.

If no file is selected or specified in the requester (and the requester is not cancelled), then Visage will display all pictures in the drawer in question (and any pattern will be used).

You may even use this argument as a tooltype in an icon. So, if you would like to get a slideshow over some pictures in a drawer, you can use the following tooltypes (in an icon without its corresponding file). The default tool should be set to Visage:

```
FILES=Pics:Nature ALL DELAY=60
```

or something similar. A double click on that icon will start the slideshow.

Note: Do not use this argument in the Visage icon to specify a default drawer, that the file requester should open in, if e.g. no other arguments were specified. The result will not be what you expect. Rather, use the DRAWER argument, which exists for this purpose.

See also: **ALL CLIPBOARD DELAY DRAWER NOFILECHECK PATTERN RANDOM SORT**

1.12 Visage.guide/Arguments/BORDERBLANK

BORDERBLANK Synonym: BB

Here you can specify if Visage should (try to) blank the borders of the displayed picture or not. The following arguments are available:

ON Turn on border blanking.

OFF Turn off border blanking.

DEFAULT Use the system default.

If an invalid blanking mode is specified, you will be told about it, and the default (DEFAULT) will be used.

Notes: This affects the systems border blanking flag only. Thus, it might not work on graphics cards (I do not know if any graphics card support this flag).

Setting the argument to OFF does not seem to work if the system global border blanking flag is activated. At least this is the case on my system.

The vertical centering does not look that good if border blanking is on.

See also: [VCENTER](#)

1.13 Visage.guide/Arguments/COLOR

COLOR Synonym: COLOUR

Here you can specify the render mode for "deep" (more than your system can handle) and true color pictures. The following modes are available:

DEFAULT, HAM The default. Pictures will be rendered in a HAM mode (HAM8 if available, HAM6 otherwise). Currently no dithering is done.

8, NORMAL Pictures will be reduced to the largest number of colors in a "normal" palette screen that is available on your system. 256 colors (eight bits) should be available, or the result will not be very good. Dithering can be used to improve quality.

16, HIGH Pictures will be rendered to screens that are up to 16 bits deep, if such screens are available via CyberGraphX (or compatible).

24, TRUE Pictures will be rendered to screens that are up to 24 bits deep, if such screens are available via CyberGraphX (or compatible).

The GRAY option will override any argument specified here.

Notes: The NORMAL color rendering is a so called one pass render with a fixed palette. This means that it is rather fast, but the quality is not the best. By default, the rendering includes ordered dither, but this may be changed, to improve either the speed or the quality (the quality will drop significantly if dithering is disabled).

The datatypes reader is also affected by this argument. If high- or truecolor rendering is requested, then Visage will ask the datatype to return data in the so called version 43 format, which is better suited for display on a graphics card screen, and includes up to 24 bit deep screens (normally, the datatypes are limited to eight bits).

See also: [DITHER GRAY Rendered pictures](#)

1.14 Visage.guide/Arguments/COMMAND

COMMAND Synonym: CMD

Here you can specify a command that can be executed for each viewed picture. To execute the command, press the `e`-key when the picture in question is displayed. When in TEST mode, the command will be executed when a picture with an error is found.

The command will be executed as if it had been started in the same drawer as the picture. The name will always be quoted (e.g. to make it easy to parse from Rexx). The command is run asynchronously, so you may continue to the next picture when you like. If the command could not be run for some reason, the screen will flash (if no command is specified, this will not happen).

By default the name of the picture will be added to the end of the command. If the string `[]` is specified in the command, then the name will replace the first occurrence of this string. This is similar to how the Alias command works (see your AmigaDOS manual for more information about Alias).

This argument makes it easy to mark pictures for deletion or something similar. In that case, a command like:

```
FileNote [] "Delete this picture"
```

can be useful. Then simply press `e` for the pictures you do not want to keep, and do the actual deletings from a shell or a directory utility. Using the above command rather than `Delete` has the benefit that it is easy to undo, in case you press the key by mistake.

The command `Move [] TO Pics:` can cause problem in case several pictures with the same name (but in different drawers) are processed. An ARexx script can be used to work around this problem, if needed. Note that a `Move` command is not included with the OS.

Note: To specify the above example from a shell, you must enter it like this:

```
COMMAND="FileNote [] *"Delete this picture*""
```

(Note the escaped quotation characters.) If you enter it as a tootype, just enter it as the first example shows.

See also: [TEST](#)

1.15 Visage.guide/Arguments/DITHER

DITHER

Here you can specify which kind of dithering that should be used, when rendering pictures to COLOR or GRAY modes. In the latter case, dither is only used when the screen displays less than 256 colors.

Currently, the following dither types are available:

NONE No dithering. This is the fastest, but also gives the lowest quality. It is not recommended if COLOR rendering is used.

ORDERED, ORD Ordered dithering. A fast dither, but the quality is not the best. Quite sufficient for grayscale images though. This is the default.

FS Floyd-Steinberg dithering. Gives very good quality, but is slower than ORDERED (about 30% in the COLOR render case).

If an invalid dither type is specified, you will be told about it, and the default (ORDERED) will be used.

See also: [COLOR GRAY Rendered pictures](#)

1.16 Visage.guide/Arguments/DRAWER

DRAWER

Here you can specify the string that should be in the drawer gadget when the file requester initially opens, and Visage is not opening a requester for e.g. a drawer specified in the FILES argument. Thus, this argument allows you to specify a default drawer for Visage, something the FILES argument does not.

See also: [FILES](#)

1.17 Visage.guide/Arguments/MODE

MODE

Here you can specify which screen mode the pictures should be displayed with. There are several ways to specify the mode:

Via a screen mode requester. Simply enter REQUEST or ? after the MODE keyword. This requires asl.library to be available.

A decimal or hexadecimal number, "describing" the mode. If the number starts with '\$' or '0x' it is assumed to be in hexadecimal. The included program GetModeID can be used to find the (decimal) number for a certain mode.

A text string (anything that is not covered by the above). Either a name from the display database, or an algorithmic name.

Specifying a (valid) mode here will override the MONITOR and WBMONITOR arguments.

If the specified mode is not capable of displaying the picture for some reason, Visage will try to find a replacement mode that is as similar as possible. Thus, you usually do not need to worry about things such as HAM and EHB. Visage should handle that automatically.

Note: Visage also understands the "normal" mode names (as shown by the ScreenMode preferences program). However, the normal mode name will only be considered if the mode name did not match an algorithmic name. Thus, e.g. "DblPAL:High Res" will use the algorithmic name, while "Multiscan:Productivity" will use the normal mode name. The NORMALNAMES switch can be used to change this behaviour (Visage will then check the display database name first).

See also: [MONITOR NORMALNAMES WBMONITOR Algorithmic mode names GetModeID](#)

1.18 Visage.guide/Arguments/MONITOR

MONITOR

With this parameter you can easily promote the pictures to display to any available monitor, providing it is capable of displaying the picture.

Simply specify the name of the monitor, or a pattern for the monitor(s), and Visage will look through the display database for any matches. If found, the picture will be promoted to use one of the specified monitors, if possible. If several monitors matched the pattern, the BestModeID() function will select the monitor it considers to be the best one to display the picture, or if they are equally good, the first "best" mode will be used.

You can also specify the special name REQUEST or ?, to get a requester containing the available monitors (except for the so called default monitor). Note that the properties window will not contain useful information, except for the frequencies for the native Amiga modes. You can currently not select more than one monitor in this requester.

Specifying a valid monitor here will override the WBMONITOR argument.

This argument is ignored if a valid MODE has been specified.

Note: The name of a monitor is usually the same as the name of the file in the 'Devs:Monitors' drawer. Remember that 'VGAOnly' is not a monitor.

See also: [MODE WBMONITOR](#)

1.19 Visage.guide/Arguments/OVERSCAN

OVERSCAN

Here you can specify the overscan type that should be used to display the picture. You can specify the following types:

TEXT Text overscan, as set by preferences. Display limits visible.

GRAPHICS, STANDARD Graphics (or standard) overscan, as set by preferences. Display limits are just out of view. This is the default.

MAX Maximal overscan. The largest overscan that the system "comfortably" can display.

VIDEO Video overscan. The largest overscan the system can display, comfortably or not.

If an invalid overscan type is specified you will be told about it, and Visage will revert to the default (STANDARD).

See also: [CENTER MONITOR NOAUTOSCROLL WBMONITOR](#)

1.20 Visage.guide/Arguments/PASSWORD

PASSWORD

Here you can specify the password for any Xpk-encrypted IFF ILBM picture found. If no password is specified, and an encrypted file is encountered, then a ReqTools requester will be opened for that file (if ReqTools is available). To disable this requester, simply enter a password! ;)

1.21 Visage.guide/Arguments/PATTERN

PATTERN

Here you can specify the string that should be in the pattern gadget when the file requester initially opens. The default pattern is `#?`.

See also: [FILES](#)

1.22 Visage.guide/Arguments/BUFSIZE

BUFSIZE

Here you can specify the size of the IO buffers (in kilobytes) used by the asynchronous IO code (used by the internal (i.e. not datatype) readers). Two buffers of roughly half the size specified will be allocated, if possible. Normally this value should not need changes, but you might get somewhat better performance with other values here (depending on the media).

Default is 8 KB. Values below 4 KB are rounded to 4 KB, but the code will (usually) try with smaller buffers, if there is not enough memory available.

Notes: This buffer size should not be larger than roughly half the average size of the files you display, or the benefit of "double buffered asynchronous IO" will be lost.

These buffers will always be allocated for a picture, even if the asynchronous code cannot be used to read it (that is, for datatype pictures. Yet another reason to keep the buffer size "reasonable". ;)

If you view a lot of JPEG pictures, you should not go below 8 KB, as this will hurt performance a bit.

1.23 Visage.guide/Arguments/DELAY

DELAY

Here you can specify the time to wait between pictures. When the timeout has expired, the next picture will be shown automatically, like in a slideshow. The delay starts counting when the currently visible picture is completely loaded. A delay of zero is valid. Default is no delay (i.e., only move to the next picture when the user requests it).

See also: [FOREVER LATELOAD LOWMEM RANDOM SORT TIMES TOFRONT WAITFORPIC](#)

1.24 Visage.guide/Arguments/FADE

FADE

Here you can specify the speed with which the pictures should fade in and out when opened and closed. 1 is the fastest, and 4 is the slowest. Values out of range are rounded to the closest valid value. Default is 0 (no fade).

Notes: HAM, PCHG and SHAM pictures can not be faded, due to the nature of these picture formats. Pictures displayed in high- or truecolor can not be faded either.

Some auxiliary memory might be needed for a successful fade. If this memory is not available, the fade will silently not be done.

1.25 Visage.guide/Arguments/GAMMA

GAMMA

Here you can specify the gamma of your screen. In order for a picture to be properly displayed on a screen, it may need some adjustment to your equipment.

Not all picture formats store the picture in a "raw" format (i.e. without adjustment to some gamma value), with information about the gamma of that data. The PNG format does, and as a result, the picture viewer needs to adjust the picture for proper display. If this is not done, a picture may appear too dark or too bright.

If you do not know what gamma your screen has, leave it at the default, 2.5, which should be fine for most normal monitors.

For further details about gamma correction, please refer to the PNG specification.

Note: Though the value is a floating point value, exponents cannot be specified, as a simplified "ASCII to float" converter is being used.

1.26 Visage.guide/Arguments/TIMES

TIMES

The number of times to view the files. Default is 1.

Any value specified here overrides the FOREVER switch.

See also: [FOREVER](#)

1.27 Visage.guide/Arguments/TOOLPRI

TOOLPRI

The task priority Visage should use when displaying the pictures. Valid range is -128 to 4 (to prevent Visage from disturbing more important tasks). Values out of range are rounded to the closest valid value. Default value depends on the starting program (usually it is 0).

This argument can be useful when Visage is used as an external screen blanker module.

Note: The Workbench automatically parses any TOOLPRI tooltype. It was added to Visage so that the priority easily can be specified in e.g. VisageOpts.

See also: [VisageOpts](#)

1.28 Visage.guide/Arguments/UNIT

UNIT

The clipboard unit to read from, if CLIPBOARD has been specified. The value should be between 0 and 255 (inclusive). Values out of range are rounded. Default is 0.

See also: [CLIPBOARD](#)

1.29 Visage.guide/Arguments/ALL

ALL

If this option has been specified, then Visage will enter any drawers encountered during the pattern matching process.

Note: If this switch is used, then you will not get a file requester if you enter the name of a drawer on the command line. Instead Visage will enter that drawer (and any drawers within) and start displaying all pictures found.

See also: [FILES](#)

1.30 Visage.guide/Arguments/BESTMODE

BESTMODE

If this switch is specified, then Visage will always search the display database for the best screen mode to display the picture with, even if it is not necessary.

Also, Visage will be less strict when looking for a replacement mode when doing monitor promotion. Visage will ignore the size of the "source" screen mode; the size will be taken from the picture instead. Only the aspect will be considered.

If a MODE has been specified, then this option will be ignored.

See also: [MODE](#)

1.31 Visage.guide/Arguments/CENTER

CENTER Synonym: CENTRE

If this switch is specified, the displayed pictures will be centered horizontally, if needed. If you also want vertical center, see the VCENTER switch.

Notes: If a screen promoter is installed, the centering may be wrong sometimes. It depends on how the screen mode is changed, and how "good" the promoting software is.

For Super72 screen modes, the centering may be more or less wrong (the amount depends on the OS version).

This centering does not seem to work on (most) graphic card screens.

See also: [VCENTER](#)

1.32 Visage.guide/Arguments/CLIPBOARD

CLIPBOARD

If this switch is specified, then Visage will try to read the data from the clipboard. Any files specified in the FILES argument are ignored.

Use the UNIT argument to specify which clipboard unit Visage should read from.

See also: [FILES UNIT](#)

1.33 Visage.guide/Arguments/COMPACT

COMPACT

If this switch is specified, then the VERBOSE output will be a bit more compact. It assumes that your console is wide enough to accommodate all output in one line. This means that the console needs to be at least 90-100 characters wide (the exact width depends on the picture and screen mode name, and if you want timing information).

See also: [QUIET TIME VERBOSE](#)

1.34 Visage.guide/Arguments/CYCLE

CYCLE

If this switch is specified, then color cycling will automatically be started for pictures that have color cycling information.

Notes: This cycling only understands "simple" (CRNG) cycling right now. I think I know how to implement the "complex" (DRNG) cycling, but I have not worked out all details yet. If you have some example code or similar, please send it to [me](#) . ;)

The cycling is done in a high priority subtask. The vertical blanking interrupt is not used, since the function used to change colors may not be called from an interrupt (it can cause deadlocks or crashes). This can make the cycling a bit jerky at times. This can not be avoided.

1.35 Visage.guide/Arguments/DATATYPES

DATATYPES Synonym: DT

Try to view the files with a datatype before trying with internal viewers. The NODATATYPES switch overrides this one, if both are specified.

See also: [NODATATYPES NOGIF NOIFF NOJPEG NOPNG](#)

1.36 Visage.guide/Arguments/ECS

ECS

Originally intended as a debugging option (to check the ECS-graphics conversion routines), I decided to leave it in. What it does is to make Visage think that only ECS graphics is available (not even any graphics card).

I read somewhere (in fidonet, I think) that some FastJPEG users (that had AA graphics) used the ECS version in grayscale mode, in order to get a fast preview of the picture. Using this switch will have the same effect. This can be especially useful on e.g. an A1200 without fast memory. It will make the rendering of deep pictures quite a bit faster.

See also: [GRAY](#)

1.37 Visage.guide/Arguments/FOREVER

FOREVER

Keep viewing the pictures until interrupted. It actually sets the TIMES argument to a very high value (4,294,967,295 :).

Any value specified in the TIMES argument overrides this switch.

See also: [TIMES](#)

1.38 Visage.guide/Arguments/GRAY

GRAY Synonym: GREY

Convert rendered pictures to grayscale.

This option will override COLOR, if both are specified.

See also: [COLOR DITHER Rendered pictures](#)

1.39 Visage.guide/Arguments/HAM

HAM

This switch tells Visage that a HAM mode should be selected for IFF ILBMs that have a depth of six bitplanes, and do not have any so called CAMG chunk (this chunk tells what kind of screen mode that should be used to display the picture). If not specified, Visage assumes an extra halfbrite (EHB) mode should be used.

Note that this switch does not force a HAM mode to be used, only that a HAM mode should be used in case of doubt.

1.40 Visage.guide/Arguments/INFO

INFO

If this switch is specified, then Visage will not display any pictures. It will simply read some information about the picture, and display that information.

Thus, all options related to how things are rendered are ignored. However, screen mode related arguments are not, as the VERBOSE option still may be used.

This switch is in many ways similar to the TEST switch. The two major differences are that the picture is not loaded at all, and that VERBOSE is considered.

This switch overrides TEST and QUIET, if they are specified as well.

Note: Datatypes will always load the entire picture, although nothing will be displayed. This can not be avoided.

See also: [MODE MONITOR TEST VERBOSE](#)

1.41 Visage.guide/Arguments/LATELOAD

LATELOAD

When viewing several files, this switch tells Visage to wait with loading the next picture until told to do so. Background loading is not disabled, only delayed. When used in slideshows, the extra memory needed for background loading is therefore delayed to when it is really needed.

See also: [LOWMEM WAITFORPIC](#)

1.42 Visage.guide/Arguments/LOWMEM

LOWMEM

If this switch is specified, then the background loading of pictures is disabled.

Note: Visage will automatically close any displayed picture, if there is not enough memory to load the next one in the background. Thus, this switch should not be needed much.

See also: [LATELOAD](#)

1.43 Visage.guide/Arguments/NOAUTOSCROLL

NOAUTOSCROLL Synonym: NOAS

If this switch is specified, then the autoscrolling of pictures is disabled. The cursor keys can still be used to move around in the picture.

Usually, when the mouse pointer reaches the edges of a screen that is larger than the display area, the screen will automatically start scrolling, to show the hidden areas.

See also: [Viewing keys](#)

1.44 Visage.guide/Arguments/NOBUSY

NOBUSY

If this switch is specified, then no busy pointer will be shown when a picture is loaded (in the foreground or the background).

See also: [POINTER Viewing keys](#)

1.45 Visage.guide/Arguments/NOCLICK

NOCLICK

If this switch is specified, then Visage will ignore any mouse click on a picture. To view the next picture, or exit Visage, the keyboard need to be used (or you could send a so called break signal).

1.46 Visage.guide/Arguments/NODATATYPES

NODATATYPES Synonym: NODT

If this switch is specified, then Visage will not try to use datatypes if the picture format was not known to Visage.

This switch overrides DATATYPES, if both are specified.

See also: [DATATYPES NOGIF NOIFF NOJPEG NOPNG](#)

1.47 Visage.guide/Arguments/NOENV

NOENV

If this switch is specified, then Visage will not consider the options stored in the configuration file or the environment variable VisageOpts.

See also: [VisageOpts](#)

1.48 Visage.guide/Arguments/NOFILECHECK

NOFILECHECK Synonym: NOFC

If this switch is specified, then Visage will not check if files specified in the FILES argument or a file requester exists before processing. By default, Visage will check if the file really exist before attempting to parse any patterns in the name.

This behaviour is needed, since at this stage Visage does not know if the file was specified in the FILES argument or if it was a multiselected icon, as a result of the generic argument parsing functions used. To work around this problem, Visage will check if the name actually refers to an existing file.

Since this behaviour can cause some minor problems (i.e., a pattern is not recognized as expected), you have this switch to decide how Visage should behave.

See also: [FILES](#)

1.49 Visage.guide/Arguments/NOFLICKER

NOFLICKER

If this switch is specified, then Visage will not use laced screen modes. Mostly useful if you have AA graphics, and you do not promote the pictures to either the NTSC monitor or the PAL monitor.

See also: [MODE MONITOR](#) [WBMONITOR](#)

1.50 Visage.guide/Arguments/NOFOREIGN

NOFOREIGN

If this switch is specified, then Visage will make sure it doesn't use certain display modes, namely those with the so called foreign flag set.

There is a certain confusion regarding this flag and its use. Because of that (and the fact that different Intuition emulators behave differently in certain aspects), Visage can get into problems, depending on which Intuition emulator that is in use. Most notably, Visage can crash when used with e.g. the PicassoII.

If you experience this problem, then this switch should (hopefully) solve your problems.

1.51 Visage.guide/Arguments/NOGIF

NOGIF

This option is always considered to be set, since Visage does not have internal support for GIF (as it used to have). As a result, GIF pictures will always be passed on to the datatypes for processing. Its presence is for backward compatibility only.

See also: [DATATYPES](#) [NODATATYPES](#) [NOIFF](#) [NOJPEG](#) [NOPNG](#)

1.52 Visage.guide/Arguments/NOICON

NOICON

If this switch is specified, then any file encountered that ends with `.info` will be ignored. This makes it easy to avoid icons, when using a (complex) pattern. This switch also affects the file requester, and prevents icons from being showed, regardless of the pattern used.

Note: As there are datatypes that can display icons (i.e. files typically ending with `.info`), you should not use this switch when you intend to view icons.

See also: [ALL FILES INFO SHOWINFO](#)

1.53 Visage.guide/Arguments/NOIFF

NOIFF

Try to view IFF files with a datatype before trying with the internal viewer. The `NODATATYPES` switch overrides this one, if both are specified.

See also: [DATATYPES](#) [NODATATYPES](#) [NOGIF](#) [NOJPEG](#) [NOPNG](#)

1.54 Visage.guide/Arguments/NOJPEG

NOJPEG

Try to view JPEG/JFIF files with a datatype before trying with the internal viewer. The NODATATYPES switch overrides this one, if both are specified.

See also: [DATATYPES NODATATYPES NOGIF NOIFF NOPNG](#)

1.55 Visage.guide/Arguments/NOPNG

NOPNG

Try to view PNG files with a datatype before trying with the internal viewer. The NODATATYPES switch overrides this one, if both are specified.

See also: [DATATYPES NODATATYPES NOGIF NOIFF NOJPEG](#)

1.56 Visage.guide/Arguments/NORMALNAMES

NORMALNAMES

If this switch is specified, then the algorithmic mode names will not be used when displaying the mode names. You may still use them when specifying a mode though, although these names will have a lower priority then (see the MODE argument for more information about this).

This might be needed if you have a graphics card (unless you use CyberGraphX). Or maybe you prefer the normal names.

If RTG is specified, then this switch will be set as well.

See also: [MODE RTG Algorithmic mode names](#)

1.57 Visage.guide/Arguments/NOWARN

NOWARN

If this switch is specified, then most warning messages will not be displayed. Some warnings can be a bit annoying when Visage is started from the Workbench.

1.58 Visage.guide/Arguments/POINTER

POINTER

If this switch is specified, then a normal mouse pointer will be displayed when the busy pointer is not displayed. Normally, no mouse pointer is visible during this time.

See also: [NOBUSY Viewing keys](#)

1.59 Visage.guide/Arguments/QUIET

QUIET

If this switch is specified, then most output from Visage is disabled. Errors and similar things are still displayed. This option is automatically used when the program is started from the Workbench.

If the INFO switch is specified, then this switch will be ignored.

See also: [INFO TIME VERBOSE](#)

1.60 Visage.guide/Arguments/RANDOM

RANDOM

If this switch is specified, then Visage will display all specified and selected files in random order, rather than the (perhaps somewhat random) order in which they were found. This switch overrides SORT, if both are specified.

Note: Visage will first scan through any patterns specified, before starting to view the pictures. This may take a little while. It also means that you will not get a new file requester after having viewed the first selected files.

See also: [DELAY SORT](#)

1.61 Visage.guide/Arguments/RTG

RTG

If this switch is specified, then Visage will not make certain assumptions about the contents of the display database (i.e. that a certain flag will be set for all non native Amiga modes). This assumption makes Visage work slightly better (i.e. work faster or use less memory) when displaying pictures in a native Amiga mode.

However, (partly) due to bugs in the current system software, this assumption is not (always?) correct when a graphics card is installed, and this can lead to problems. Thus, if Visage seems to be unable to display a picture on a "graphics card screen mode", then you should try this option.

Thanks to a new way handling things (in Visage 39.2), this switch should not be needed, but I find it best to make no such assumptions! ;) Also, Visage is capable of recognizing CyberGraphX modes, so this switch will not be needed if you have CyberGraphX installed.

This switch implies NORMALNAMES.

See also: [NORMALNAMES](#)

1.62 Visage.guide/Arguments/SCALE

SCALE

If this switch is specified, then rendered pictures will be scaled down to the visible part of the screen in question, if needed.

Note: This scaling uses a simple (but fast) algorithm, and hence the result may not be that good at times. The result largely depends on the picture in question.

See also: [Rendered pictures](#)

1.63 Visage.guide/Arguments/SHOWINFO

SHOWINFO

If this switch is specified, the Visage will not only display picture information in the Shell it was started from (if any), but will also display it on (or rater, above) the actual picture.

The VERBOSE switch can be used to specify how much information that should be displayed, similar to the output in a Shell. TIME information is (currently) not displayed on the picture.

This switch is not affected by the QUIET switch.

Notes: This information is usually not displayed for datatype pictures. Rerendered datatype pictures will have this information though.

Visage will only use a single line for the text. This means that the text can be truncated, if the screen is not wide enough.

The font used is the one used by the default public screen (at the time Visage was started), or, if Visage was unable to find the default public screen, the current system default font.

Visage will examine the colormap, and will try to find the best colors to display the text. However, the text may not always be (clearly) visible anyway. Also, when used with PCHG or SHAM pictures, the color selection may seem a bit odd (and incorrect).

See also: [QUIET TIME VERBOSE Rendered pictures](#)

1.64 Visage.guide/Arguments/SORT

SORT

If this switch is specified, then Visage will display all specified/selected files in alphabetical order, rather than the (perhaps somewhat random) order in which they were found. The RANDOM switch overrides this switch, if both are specified.

Note: Visage will first scan through any patterns specified, before starting to view the pictures. This may take a little while. It also means that you will not get a new file requester after having viewed the first selected files.

See also: [DELAY RANDOM](#)

1.65 Visage.guide/Arguments/TEST

TEST

If this switch is specified, then Visage will not actually display the pictures. Rather, it will try to load them to see if they seem to be free from errors. When all files have been tested, Visage will print out a status message, showing if any picture contained any error. Also, if there was any error, the WARN flag will be set.

For each file that failed with a "data" error, an error message will be printed, and the COMMAND will be executed (if a command was specified). Errors caused by external things (such as out of memory) are not considered here.

If this switch is specified, then many other switches and arguments will be ignored.

If INFO is specified as well, then this switch will be ignored.

Notes: Only certain errors causes Visage to treat it as a bad. I.e. exit with the WARN flag and execute the command.

Even if a picture "fails", it might still be more or less viewable (e.g., it is not a major problem the error occurred while loading the last few pixel lines of the picture). Conversely, the fact that a picture passes the testing does not mean that the picture is free from trashed graphics.

It seems like most datatypes does not return any error, even if the picture contained detectable errors. Thus, you should use the NODATATYPES option to make sure that this does not happen.

See also: [COMMAND NODATATYPES INFO](#)

1.66 Visage.guide/Arguments/TIME

TIME

If this switch is specified (and QUIET is not), then Visage will display how long it took to load (and possibly render) the picture. This time does not include the time it took to open the screen in case a datatype picture was loaded. This makes a difference for some datatypes (that does not return a so called displayable bitmap), and when the picture is displayed on a graphics card screen.

Note: Since the QUIET option always is on when Visage has been started from the Workbench, there is no point in specifying this option in the tooltypes of an icon.

See also: [QUIET VERBOSE](#)

1.67 Visage.guide/Arguments/TOFRONT

TOFRONT

If this switch is specified, then the picture will be brought to the front each time a new picture is to be displayed. This is useful in slideshows in screen blankers, which should bring their screens to the front every now and then (in case some other program has opened a screen in front of the blanker).

See also: [DELAY RANDOM SORT](#)

1.68 Visage.guide/Arguments/VERBOSE

VERBOSE

If this switch is specified (and QUIET is not), then some extra information about the pictures will be displayed. Currently this is the screen size and mode used by the picture. This screen mode name uses the algorithmic name, unless the NORMALNAMES switch has been specified.

If the SHOWINFO switch has been specified, the extra information will be displayed on the picture as well.

Note: Since the QUIET option always is on when Visage has been started from the Workbench, there is no point in specifying this option in the tooltypes of an icon, unless SHOWINFO has been specified.

See also: [NORMALNAMES QUIET SHOWINFO TIME Algorithmic mode names](#)

1.69 Visage.guide/Arguments/VCENTER

VCENTER Synonym: VCENTRE

If this switch is specified, the displayed pictures will be centered vertically, if needed.

Notes: If a screen promoter is installed, the centering may be wrong sometimes. It depends on how the screen mode is changed, and how "good" the promoting software is.

Due to the implementation, it does not look that good if border blanking is enabled (this can not easily be avoided).

The centering:

Only works for datatype pictures that need to be re-rendered.

Will make the picture use some more graphics memory.

Should work on all graphic cards (that put proper information in the display database).

See also: [CENTER](#)

1.70 Visage.guide/Arguments/WAITFORPIC

WAITFORPIC

If this switch is specified together with DELAY - and perhaps LATELOAD - then Visage will wait for the next picture to load completely before switching to that one (if there is enough memory to load the picture in the background).

Visage will not wait if you tell it to move to the next picture.

See also: [DELAY LATELOAD LOWMEM](#)

1.71 Visage.guide/Arguments/WBMONITOR

WBMONITOR Synonym: WBMON

Similar to MONITOR, but it will promote the pictures to the same monitor as the Workbench screen is using (if possible). The MONITOR argument overrides this switch, if both are specified.

Note: If you use the so called default monitor for your Workbench screen, then this option will not work. This can only happen if the relevant preferences file is not available for some reason (to create it, simply select a suitable mode in the ScreenMode preferences editor and select "Save" or "Use").

See also: **MODE MONITOR**

1.72 Visage.guide/Viewing keys

Viewing keys

When displaying a picture, there are several keys you can use:

Esc, Q Quit Visage.

E Execute the **COMMAND**, if specified.

B, N Toggle the **NOBUSY** option.

M, P Toggle the **POINTER** option.

Space, Return, Enter View the next picture. If the last picture is shown, show a file requester or exit.

You can also use the cursor keys or the numeric keyboard to scroll around in the picture. When doing this, no qualifier will scroll using steps of 10 pixels, Shift uses steps of a quarter of the visible size, Alt uses steps of three quarters of the visible size, while Control moves to the extreme.

The mouse can be used as well (unless the **NOCLICK** switch has been specified):

Left button View the next picture. If the last picture is shown, show a file requester or exit.

Right button Quit Visage.

You can of course also use these keys/mouse buttons while a picture is loaded. There may be a small delay before Visage reacts though (this is usually due to IO buffering, or a datatype picture is being loaded).

1.73 Visage.guide/Algorithmic mode names

Algorithmic mode names

An algorithmic name is a name that is built out of information available in the so called display database. These names are rather similar to the (English) names seen in e.g. the ScreenMode preferences program, but some things differ.

But why have different names? Well, the reason is rather simple actually. The display database only contains names for relatively few screen modes¹. With these algorithmic names, you can specify any mode using text (in a consistent manner). You are not limited to the rather few modes the OS programmers decided to put names on.

Mode name parts How a mode name is "constructed". **Mode name examples** Some examples. **Mode name notes** Some notes about the mode names.

Visage also supports the more traditional (and in some ways simpler) way of displaying modes without a name, where the name consists of the monitor name, the screen size, and any special mode flags. Examples:

```
PICCOLO:1024x768 MULTISCAN:320x240 HAM
```

-- Footnotes --

¹ By using a custom sys/monitors.catalog file, more names can be added. The problem is that screen mode names are limited to about 30 chars, and this is not enough for many modes (especially MULTISCAN modes). The algorithmic names can be much longer (the internal buffer is currently 256 chars ;).

1.74 Visage.guide/Algorithmic mode names/Mode name parts

Mode name parts

A name is made of a couple of components, of which some are required, and others are optional. The different components are separated from each other by a single space, except for the first two.

This is the how the mode name should look (required components enclosed in <>, optional components enclosed in []):

<Monitor name:><Horizontal resolution> [Vertical resolution] [Special mode flag]

And here is an explanation of the different components:

Monitor name: The name of the monitor, including an ending colon (:). Example: "PAL:".

Horizontal resolution: The nominal horizontal size (i.e. without overscan). Use one of the following: Extra-Low Res Less than 200 pixels wide. Low Res Between 200 and 400 pixels wide. High Res Between 400 and 800 pixels wide. Super-High Res More than 800 pixels wide.

Vertical resolution: The nominal vertical size (i.e. without overscan). Use one of the following: Double So called doublescan mode. Usually less than 200 pixels high. (Nothing.) Normal mode. Usually between 200 and 400 pixels. Sometimes more, up to 5-600 pixels. Laced Interlaced ("flicker") mode. At least 400 pixels. Often more.

Special mode flag: Use one of the following: HAM Hold And Modify. A special "compressed" mode with many colors, but color selection is restricted. Six or eight bitplanes. EHB Extra HalfBrite. 64 colors (six bitplanes). The second half of the colors are copies of the first, but they are half as bright. There are a few more special mode flags that could be used, but they are not useful in this application.

1.75 Visage.guide/Algorithmic mode names/Examples

Mode name examples

Here are some examples on how the algorithmic mode names can look (English system defaults in parentheses):

DbIPAL:Low Res (DbIPAL:Low Res No Flicker) Euro72:High Res (Euro72:Productivity) Euro72:Extra-Low Res Double («No equivalent») DbINTSC:Low Res Laced (DbINTSC:Low Res Laced) Super72:Super-High Res Laced HAM («No equivalent»)

Please note that e.g. the modes `PAL:Low Res` and `DbIPAL:Low Res` have a different aspect with these kind of names! The "equivalent" of `PAL:Low Res` in DbIPAL is `DbIPAL:Low Res Double`.

1.76 Visage.guide/Algorithmic mode names/Notes

Mode name notes

It is usually not that important to specify the right special mode. If a special mode is needed, but not specified, then Visage will try to find a suitable replacement mode.

The strings will be translated if a suitable catalog is installed and used, but Visage will always understand the English strings listed above. Also note that the monitor name is never translated.

The algorithmic names are only suitable for native Amiga modes. Visage is currently only capable of detecting CyberGraphX modes, and will use a name consisting of the monitor name, width, height and depth (e.g. "CYBERGRAPHX:800x600x16"). For graphics cards with other emulation software, I recommend the use of the NORMALNAMES switch.

See also: [Mode name parts NORMALNAMES](#)

1.77 Visage.guide/Rendered pictures

Rendered pictures

A rendered picture is simply a picture that Visage must do more with, apart from reading - and possibly unpacking - the data, in order to be able to display it. Some picture formats (e.g. JPEG and PNG) are not stored in the normal Amiga bitmap format, and must therefore be converted. Others are in bitmap form, but your Amiga might not be able to display them "as is" anyway (e.g. 24-bit IFF ILBM pictures). These must be converted as well.

The rendering of such pictures can be controlled by using the `COLOR`, `GRAY`, `SCALE` and `DITHER` options (and the `ECS` option).

Datatype pictures are a special case: If the system is not capable of correctly displaying a datatype picture, Visage will re-render it to a displayable format. If the system is capable of showing it, no re-render will be done, as this process costs rather much time and memory.

CyberGraphX

Things become a bit different if CyberGraphX is installed. Visage should be able to display pictures without the above mentioned "rendering" (rather, it is bitmapped pictures that need re-render for CyberGraphX screens).

Despite this, Visage will by default not use CyberGraphX. Deep pictures (see above) will still be rendered in HAM format. To change this, use the `COLOR` argument.

See also: [COLOR DITHER ECS GRAY SCALE](#)

1.78 Visage.guide/Included programs

Included programs

In this archive (in the drawer ``Visage/C``) you can find a couple of extra support programs that I have written (some of these can also be found in the PicBoot archive):

DeleteLink Delete files without checking them first. **GetModeID** Print the mode id number for a screen mode. **Kill** Stop another task, similar to `Break`. **MakeLink** Make links, supports soft links too. **UnpackILBM** Unpack the `BODY` chunk of an IFF ILBM file.

The following program is used during the installation:

UpdateTT Update tooltypes from one icon to another.

Feel free to use the programs above wherever you like, but if you do copy them, distribute them with another program or anything like that, please include the manual page in question, and a note that I wrote them.

Finally, in the ``Visage/Rexx`` drawer, you can find an ARexx script written by Leo Davidson. Its purpose is to use Visage to show slideshows in Directory Opus 5. There are two versions of this script, for different versions of Directory Opus.

1.79 Visage.guide/Included programs/DeleteLink

NAME

DeleteLink -- Delete files without checking them first.

TEMPLATE

DeleteLink FILES/A/M,QUIET/S

SYNOPSIS

DeleteLink [FILES] <Files...> [QUIET]

DESCRIPTION

Simply try to delete the specified files. DeleteLink will not try to find out if it is possible to delete the file or anything like that. Patterns are not supported.

OPTIONS

FILES The files that should be deleted.

QUIET Do not print the names of the files as they are deleted.

NOTES

The normal Delete command in AmigaOS can not delete a soft link, if the target for the soft link has been deleted. The Delete command tries to "lock" the file to be deleted, in order to get some information about it (e.g. if it is a drawer or not). For soft links, this locking also means that the link will be resolved; i.e. the OS tries to lock the link target, as well as the link itself. If the target file doesn't exist, the locking fails. As a result, Delete thinks that there is no file to delete, and quits.

DeleteLink avoids that problem by "blindly" trying to delete the file, without first locking it.

1.80 Visage.guide/Included programs/GetModeID

NAME

GetModeID -- Get screen mode id number.

SYNOPSIS

GetModeID

DESCRIPTION

Open an Asl or ReqTools screen mode requester, and print out the decimal number for the selected screen mode. Only the number is printed, making it suitable for "backticking" it into the command line of other programs.

EXAMPLE

A bit useless perhaps, since Visage has a mode requester (among other things) already, but just an example to show how it can be used:

```
Visage Pics:Some.Pic MODE `GetModeID`
```

1.81 Visage.guide/Included programs/Kill

NAME

Kill -- Send a break signal to a program.

TEMPLATE

Kill NAME/A,ALL/S,C/S,D/S,E/S,F/S

SYNOPSIS

Kill [NAME] <Name> [ALL] [C] [D] [E] [F]

DESCRIPTION

Send the specified break signal(s) to the specified program. If the program is not found, do nothing and return WARN. Any combination of break signals are allowed.

Exactly how a program reacts (if at all) to a certain break signal is up to the program, but below you can see the typical action (if any is "defined").

OPTIONS

NAME The name of the program that should receive the break signal. Note that this is the so called process name, not the program name. They might differ, but usually they should not. The first program with this name is signalled. The search is not case sensitive.

ALL All break signals below are to be sent.

C Send the break signal Ctrl-C. The program usually quits upon receiving it. This is the default signal.

D Send the break signal Ctrl-D. Some programs (such as Visage) abort the current "operation" (or file), and proceeds with the next (if any).

E Send the break signal Ctrl-E.

F Send the break signal Ctrl-F. The program usually "pops up" (bringing any window to the front) upon receiving it.

1.82 Visage.guide/Included programs/MakeLink

NAME

MakeLink -- Make a link from one file or drawer to another.

TEMPLATE

MakeLink FROM/A,TO/A,HARD/S,SOFT/S,FORCE/S

SYNOPSIS

MakeLink [FROM] <Source> [TO] <Dest> [HARD] [SOFT] [FORCE]

DESCRIPTION

Creates a file on a disk that is a pointer to another file. When an application or command calls the FROM file, the TO file is actually used. By default, MakeLink makes hard links - the FROM and TO files must both be on the same volume.

To make soft links - which can cross volumes - the SOFT keyword must be specified. This might not be supported by all filesystems or OS versions.

Normally, MakeLink does not support drawer links, as they can be dangerous to applications. To create a drawer link you must use the FORCE option. If MakeLink detects that you are creating a circular link, such as a link to a parent drawer, you will receive a Link loop not allowed message.

OPTIONS

FROM The name of the link to be created.

TO The file/drawer the link should point to.

HARD Create a hard link. Hard links can only point to files or drawers on the same volume. This is the default.

SOFT Create a soft link. Soft links can point to files or drawers on other volumes.

FORCE You must specify this in order to make a drawer link.

NOTES

MakeLink requires OS 3.0 or better to operate, as soft links are said to not be fully supported in earlier OS versions.

OS 3.0 seems to only allow the creation of soft directory links. OS 3.1 supports soft file links as well.

Not all programs support soft links. Especially soft file links can cause problems, and may for example show up as a directory.

Using soft links can be a bit tricky. If you delete the file/directory the soft link points at, then not all programs will be able to delete it. In that case, you have to create the file the link points to, and then delete the link (and finally the file). Or you could use the included [DeleteLink](#) program.

The complete path to soft links (including the file name) can not be more than 256 chars long, due to limitations in the current OS version. If you try to create the link anyway, you will get a nonfunctional link.

1.83 Visage.guide/Included programs/UnpackILBM

NAME

UnpackILBM -- Unpack IFF ILBM pictures.

TEMPLATE

UnpackILBM FROM/A,TO

SYNOPSIS

UnpackILBM [FROM] <Source> [TO <Dest>]

DESCRIPTION

Unpack the so called BODY chunk of an IFF ILBM file. The BODY chunk contains the actual picture data. The other parts of the file are simply copied.

OPTIONS

FROM The picture to unpack.

TO The file to unpack to. If not specified, then the unpacked file will overwrite the original (via a temporary file). Specifying the same file for both FROM and TO causes an error.

FUTURE PLANS

Several things could be improved, including buffered I/O (and thus not loading the entire BODY chunk into memory and unpack it there); a chunk filter (to remove unwanted chunks). Other (less likely to be implemented) things include the possibility to add/change some chunks and pack the BODY chunk.

1.84 Visage.guide/Included programs/UpdateTT

NAME

UpdateTT -- Update tooltypes from one icon to another.

TEMPLATE

UpdateTT FROM/A,TO/A,TT=TOOLTYPES/A/M

SYNOPSIS

UpdateTT [FROM] <Source> [TO] <Dest> [TOOLTYPES] <ToolType> ...

DESCRIPTION

Update tooltypes from one icon to another. The tooltypes specified (patterns are allowed) that are found in the FROM icon, will be copied to the TO icon, but only if they did not already exist in the TO icon. All other tooltypes are left unchanged.

When specifying the tooltypes, only the actual "name" of the tooltype is considered. Any arguments are ignored, as are "comment parentheses" ("()") around the tooltype to disable it).

OPTIONS

FROM The name of the icon to read the tooltypes from, without the ".info" extension.

TO The name of the icon to update the tooltypes to, without the ".info" extension.

TOOLTYPES The tooltypes that should be updated. Patterns are allowed. Parentheses around tooltypes in the icons (to disable them) are ignored, as are any arguments.

1.85 Visage.guide/Notes

Notes

Visage has been extensively tested on a variety of different Amigas (sometimes with Enforcer and/or Mungwall running), and I am happy to say that there are not many known problems at the moment.

However, there are a couple of things that you should be aware of:

Back picture visible **Bad aspect**

Graphic cards **Interlaced JPEG and PNG pictures**

Mouse jumping **Pipe reading**

PNG support **Screen centering**

Tooltips ignored

There is also one bug that I have been unable to fix. Luckily, it is harmless. Sometimes, the **fade** will stop when the picture is very dark (but not completely black). I do not know why or when this happens; I have only noticed it once.

I would not be the least surprised if more serious problems or bugs turns up. After all, bug free software is a goal that is impossible to reach. :)

1.86 Visage.guide/Notes/Back picture visible

Back picture visible

Under some very special circumstances, the picture loaded in the background may be partially visible. This cannot be avoided, and is harmless.

1.87 Visage.guide/Notes/Bad aspect

Bad aspect

Some pictures have a strange aspect stored in the file. Thus, if you do get a strange aspect for a picture, this is the most likely explanation (or maybe your Amiga does not have a screen mode with a proper aspect).

If you have a graphics card, it is possible that the display database contains improper values (for the modes provided by the graphic card). The internal BestModeID() function is very strict when it comes to the aspect, so this should not be the source of the problem.

1.88 Visage.guide/Notes/Graphic cards

Graphic cards

As I only own one graphics card, it is a bit tricky to ensure that Visage really works with all the different "Intuition emulators" available.

One important thing to remember is that you should use the **COLOR** argument, to change how deep pictures should be rendered, and possibly the **DITHER** argument as well.

Here is a list over how the most common emulators behaves (according to the reports I have got! ;):

EGS Pictures are displayed properly. However, it seems like the background loading does not work. The **LOWMEM** or **LATELOAD** option should be specified in that case. No high- or truecolor render.

Picasso No known serious problems. Possibly some trashed graphics at the right side of the picture (but hopefully not). Please note that the **NOFOREIGN** switch might be needed for Visage to work properly (i.e. not crash ;). No high- or truecolor render, unless the Picasso96 CyberGraphX API is installed. (Note that I have not tested Picasso96 myself.)

CyberGraphX The current beta of version 3 works fairly well, though you should avoid using the CENTER option, as this (new) part does not work that well yet. Specifying LOWMEM reduces the problems, but does not completely remove them. Version 2 should work as well, but I have not tested this myself.

I have not been able to trace any of the above problems to bugs in Visage (maybe some of the problems can be avoided, if I only knew how ;). It should be noted that other versions of the emulation software may work differently.

See also: **CENTER COLOR DITHER LATELOAD LOWMEM NOFOREIGN NORMALNAMES RTG**

1.89 Visage.guide/Notes/Interlaced PNG and JPEG pictures

Interlaced PNG and JPEG pictures

PNG and JPEG both support "interlaced" pictures (the JPEG term is progressive), where the picture is first drawn very roughly, with the details being filled in later. In order to do this, the picture is rendered in several passes, where each pass increases the displayed detail level.

Visage will render these pictures this way, but since the rendering process can be rather slow - especially for truecolor pictures rendered in HAM - only every other display pass will actually be rendered. This means that there can be a (small) delay between two rendering passes (especially for the final pass for PNG pictures). On the other hand, the total render time can decrease significantly.

Also, this kind of interlaced pictures can require quite a bit of memory to load. This especially true for PNG pictures, since a full size copy of the picture data - in uncompressed form - must be available during load, apart from the actual screen data.

So, if you have a truecolor picture of, say, 640×480 pixels, Visage needs an image buffer of $640 * 480 * 3 = 933,120$ bytes, plus some additional overhead. This image buffer currently needs to be in one continuous memory block.

JPEG pictures needs some extra memory as well, but I do not know how much, nor how it is handled internally.

1.90 Visage.guide/Notes/Mouse jumping

Mouse jumping

During picture loading (especially in the background), the mouse pointer "jumps" a little horizontally. I do not know if this is specific to my computer (and all my background utilities), but it does happen for other picture viewers as well.

I do not know why this happens; if it is due to a bug in Visage (not likely, considering that other programs have similar "problems", and it is only visible for native Amiga screen modes) or what. It seems to be harmless though.

1.91 Visage.guide/Notes/Pipe reading

Pipe reading

Most of the time, Visage should be able to read and display data from a (named) pipe. However, there are a couple of side effects you should be aware of:

1. Pipe reading does not work for datatype pictures, but will probably work for Xpk-compressed pictures (but not PowerPacked pictures), if xpkmaster.library version 3 or higher is installed. There is nothing that can be done about this at the moment. As a result, it could be a good idea to use the NODATATYPES switch, just in case, if you want to use this feature.
2. Visage will always keep the asynchronous buffers allocated, even if they are not used. This is not really possible to do something about, due to some internal details and interactions between some of the arguments available.
3. As a result of the above, you should not use very large values for the buffer size (the BUFSIZE argument), as it would waste memory for the picture types mentioned in point one. You should not have very large buffers anyway (say, more than 64 KB or so), as they cost memory, and does not give very good results anyway.

See also: **BUFSIZE NODATATYPES**

1.92 Visage.guide/Notes/PNG support

PNG support

Visage should be able to display just about any valid PNG picture available. It also supports every important part of the PNG specification, including gamma correction, transparency handling and alpha channel processing. There is only one omission worth mentioning:

Text information PNG allows various kinds of text information to be stored with the picture, and this information should be displayed when the picture is viewed. For now, Visage will ignore this.

Another thing worth mentioning is that the creation of the gamma correction tables for PNG pictures with 16 bits per channel can take a fairly large amount of time. However, such pictures should not be that common anyway. ;) Also, this delay could probably be removed, but it requires (fairly large) changes in libpng.

1.93 Visage.guide/Notes/Screen centering

Screen centering

The screen centering algorithm does not seem to work most "intuition emulators" for graphic cards (I think I know how this could be worked around, but I have not bothered with it yet ;). Strictly speaking, the emulation software does not handle a feature in AmigaOS.

Also, the centering may be more or less wrong for Super72 screens. The exact amount depends on the OS version.

1.94 Visage.guide/Notes/Tooltypes ignored

Tooltypes ignored

It is possible that the tooltypes specified in the Visage icon are ignored. This can happen if you e.g. double click on an icon that has Visage as the default tool. If the Workbench needs to search the command path for Visage, it will then pass on information that makes Visage believe that the program is located in the same drawer as the project icon, which is not the case. Seems to be a Workbench bug if you ask me. ;) I do not know which versions of Workbench that have this problem. 3.1 (40.5) has it at least.

To work around this problem, select the project icon, and double click on the Visage icon with shift pressed, or make sure the project icon contains the full path to Visage.

1.95 Visage.guide/The future

The future

There are a several things that could be improved or implemented. These are things I (or a user or beta tester) have thought of during the development, but did not get implemented, either due to lack of information, time, "interest" (i.e. if I thought it could be useful) or something similar.

Anyway, here is a list over the more important ones:

Improve the quality of rendered "deep" pictures in ECS modes. Code that detects grayscale pictures could be nice. Future versions of the libpng code might help as well (the current code is too slow).

A GUI mode, to make Visage easier to use, especially from the Workbench. This would probably mean a set of menus, and slightly different behavior (when the mode is enabled).

Load more than one picture in the background (similar to FV).

And a few minor ones:

Break Visage up into modules loaded on demand. Mostly useful for the JPEG and PNG picture loaders, but it could allow for further picture formats to be added easily. Work in this direction has been started, but nothing is certain yet.

Internal multitasking. This would mean that Visage would better respond to e.g. cursor keys, while loading a datatype picture.

If you have any suggestions on what you think should be implemented, feel free to contact [me](#) .

I have no immediate plans to add the following:

Animation support. I suggest you use a dedicated animation viewer instead.

More picture formats. Maybe a few more formats will be added, but I do not know of any other right now (and using datatypes is much better for "odd" formats anyway).

1.96 Visage.guide/Acknowledgements

Acknowledgments

There are a couple of persons who have helped me in one way or another with this program (in no particular order):

The Independent JPEG Group The libjpeg link library.

The PNG Development Group The libpng link library.

Jean-loup Gailly and Mark Adler The zlib link library.

Christoph Feck The Tower JPEG Codec Class, which was used in earlier versions of Visage.

Rafael D'Halleweyn The fast 24-bit to HAM6/HAM8 encoding functions.

Sebastiano Vigna The PCHG specification and example code.

Martin Taillefer The asynchronous IO code and the FixAG program. (Although I had to fix a bug or two in the async io code myself! ;)

Wessel Dankers For the MergeList() function (as posted in comp.sys.amiga.programmer).

Günther Röhrich His ideas/suggestions, and the help he gave me about his HAM encoding function, although I decided to not use it.

Lars Eilebrecht For DoIconLite, which was used during the installation in earlier versions of Visage. I replaced it with a program written by myself, since DoIconLite did not quite work the way I wanted.

Phil Vedovatti For the nice NewIcons.

John Hendrixx For the chunky to planar conversion routines (sorry, the planar to chunky functions are not flexible enough ;).

Lee Kindness For Umentiler, and being kind (!) enough to implement my specific requests, so that I don't need to mess around with AREXX scripts and environment variables to process this manual.

Erik Lindberg · Johan Billing · Hans Bergengren · Mathias Karlsson · Mattias Johansson · Reinhard Katzmann · Thorsten Marquardt · Lars Eilebrecht · Nick Soggia · Rolf Rotvel Beta testing (during one time or another). A special thank must go to Mathias Karlsson, who suggested the name for this program.

Finally, I would like to thank everyone who has sent me a message, with bug reports, comments or similar. A very special thanks to those who have sent me a donation.

1.97 Visage.guide/Author information

Author information

If you wish to send me comments, bug reports, gifts, money or whatever, then use this address:

Magnus Holmgren Kvarnbergsvägen 5 S-444 47 Stenungsund SWEDEN

Electronic addresses:

Fidonet: 2:203/512.10 Internet: cmh@lls.se WWW: <http://www.lls.se/~cmh/>

The latest version of Visage is available for download from my home page. There you can also find some information about me, and other programs I have written.

If you send me a bug report, please read the relevant sections of this manual first (mainly the **argument** section if suitable, and the **notes** section). Many known "problems" (or features ;) are mentioned here.

Also, please include some basic system information with your bug report (OS version, CPU type and graphics subsystem are of main interest here). If the problem only occurs in certain cases (e.g. a certain picture type or size, or certain arguments), please specify this as well. Thank you.

1.98 Visage.guide/Program history

Program history

[Version 39.0](#) [Version 39.1](#)

[Version 39.2](#) [Version 39.3](#)

[Version 39.4](#) [Version 39.5](#)

[Version 39.6](#) [Version 39.7](#)

[Version 39.8](#) [Version 39.9](#)

[Version 39.10](#) [Version 39.11](#)

[Version 39.12](#)

1.99 Visage.Guide/Program history/Version 39.0

Version 39.0

Initial release.

1.100 Visage.Guide/Program history/Version 39.1

Version 39.1

Various bug fixes to the Xpk support code (it was a last minute addition. The beta testers never saw it ;):

Certain seeks did not work. Usually caused crashes if the unpacking was broken.

After a break, the next picture will be started faster (never noticed, due to the bug above ;).

Errors (during read) was not quite properly handled.

Fixed a bug that caused MungWall hits, but usually nothing else. ;)

Tried to remove the "odd" error requesters some users got after cancelling the file requester. I am not sure why this happened, but hopefully the changes will help. If not, I guess I will have to remove it, although this must be due to some "bad" Asl patcher or something similarly buggy. ;)

Corrected the version string. The last part of the version number should never have been there.

The working name of the program (PicView) was left in a few places. Funny, I thought I had searched through the relevant files.. ;)

Please note that my Fidonet address changed to 2:204/204.6 on the 13th of August ('95)!

Added support for color rendering of pictures, based on code from the Independent JPEG Group. The quality is not great perhaps (largely based on the fact that the palette is fixed), but on a 256 color screen the result is fairly good, at least with dithering.

Now (optionally) **dithers** grayscale **rendered** pictures on ECS Amigas. Improved the quality quite noticeably.

Made various minor cleanups to the program. Hopefully made it a bit smaller. ;) (Eaten up by all additions..)

Updated the installer script a bit.

Norwegian translation by Jan-Tore Eliassen included.

Included a simple AReXX script written by Leo Davidson. Its purpose is to use Visage for a slideshow in DirOpus5.

Removed an Enforcer hit. Only occurred when Visage encountered a file it could not display, and no other picture was displayed.

Added support for soft file links (the OS handles soft dir links by itself).

Included a **MakeLink** clone I wrote quite a while ago, but never released, so that you can make soft links yourself. Note: This clone has the same size as the original (MakeLink 37.4)! ;)

Fixed a bug that (probably) could cause the wrong filename to be displayed sometimes. I never noticed it... ;)

Included DoIconLite by Lars Eilebrecht, to allow easy and flexible update of the tooltypes in the icon (in the installation script).

Added the **TIME** tooltype. So now you can see how long it took to load each picture. Note: This is another "last minute addition", so do not be surprised if things does not quite work as expected. ;)

Added **COMPACT** switch. By default, Visage is now displaying verbose output on a separate line, to make it look better on consoles with a "limited" width. Using this switch will revert to the old one-line format.

If an error occurred when using the **RANDOM** option, some memory (and locks on drawers) was not always freed.

Updated the asynchronous IO code, to work around a problem when using SnoopDos with "ShowReXX" active. Caused enforcer hits if Enfocer and Mungwall was running.

1.101 Visage.Guide/Program history/Version 39.2

Version 39.2

Compile date: November 7, 1995 Program size: 46312 bytes

Finnish translation by Osma Ahvenlampi included.

The GIF reader sometimes reported odd errors if there was an error in the picture.

If there was an error in a picture, the error message was printed before the picture information.

No newline was printed after the error message.

Timing information output was not quite right (leading parenthese was shown when it should not, and vice versa).

Empty files now get a meaningful error message.

Did some changes to improve RTG support (i.e. an attempt to make it actually work ;).

Removed an Enforcer hit when using **GRAY** output and **DITHER NONE**.

Added the **NORMALNAMES** switch, in case you prefer the normal display mode names (and sizes plus modes for those without a name), or you have a graphics card.

Fixed a typo in this manual, so that one link to the man page for DoIconLite works.

Certain constructed mode names were not correct. Could only happen for "foreign" (normally graphics card) modes without a name. Made a couple of other changes to the related code.

Implemented color cycling. Added the **CYCLE** argument, to automatically start the cycling when needed. This cycling only supports "simple" (CRNG) cycling at the moment.

SORT argument added. If specified, then the pictures will be displayed in alphabetical order. Behaves otherwise as **RANDOM**.

NOCLICK argument added. If specified, then Visage will ignore any mouse clicks on the pictures. To view the next picture or exit, the keyboard need to be used.

COMMAND argument added. Here a command can be specified, that is executed when `e` is pressed.

VisageOpts may now contain comments. Comments starts with a `#` or `;` on the beginning of a line, and ends with the end of that line.

Visage now looks for the file VisageOpts in the drawer specified in the environment variable ConfigPath, before looking for the environment variable VisageOpts.

TEST argument added. Causes pictures to be loaded, but not displayed. If a picture fails, the **COMMAND** is executed, if specified.

Fixed a few problems in the installer script.

Cleaned up the output code. The WShell scrollbar looks much better now.

If you specify a file in the file requester that refers to a file that does not exist, then Visage will interpret it as a pattern. This allows you to select a drawer, and then specify a pattern in a very easy way. ;)

Replaced some catalogs strings.

Rewrote dithering functions into assembler. Overall render speed increased some 13-18 percent (depending on dither mode) on a rather large (scaled) JPEG on my system (A4000/040).

Added the **RTG** switch. Forced by the fact that Visage assumed a certain flags in the display database would be set for foreign screen modes, but due to bugs in some system software, this was not the case. This switch will make Visage slower on rendering some pictures, and/or use more memory for datatype pictures, when displayed in native Amiga modes. Thus, it should only be used when needed; i.e., when a graphics card is installed.

Visage will now always try to use the aspect of the screen mode stored in an IFF ILBM picture. This did not always happen earlier.

Visage did not render dithered grayscale images properly (the picture was almost completely black).

Added some simple CyberGraphX support. Visage is now able to properly recognize a CyberGraphX mode, and will behave accordingly. Thus, the **RTG** switch is not needed if you have CyberGraphX installed.

Replaced the NewIcon image with one drawn by Phil Vedovatti.

Visage sometimes exited without reason when a bad file was found.

Changed to chunky <-> planar conversion code written by John Hendrixx (except for chunky to planar for 8-bit screens). Shaved some 700 bytes from the program, and made it a tad faster.

If no password was specified, but one is needed, Visage will now prompt for one, if ReqTools is available.

Added GRAPHICS as a synonym to STANDARD, for the **OVERSCAN** argument.

Added some safety checks.

Cleaned up some parts of this manual. Also changed the layout a little.

Reduced memory usage during dither and/or color render (at most a couple of KBs).

Added vertical centering (**VCENTER**). Note that this looks best if border blanking is off, due to how it is implemented. It also makes the picture use more graphics memory.

Fixed a bug that could make gray scale rendering a tiny bit slower than needed. ;)

Added support for background color in GIFs (of version 89a). Nice when the border blanking is not on, and/or vertical screen centering is used. Note that not all GIFs use this feature.

Localized date in "initial about string". ;)

Changed way of rendering interlaced GIFs. Looks much better, but is slightly slower. This method is not used if the picture is scaled (it will not look good.. ;).

Changed way of detecting foreign modes. Not as flexible as the old one, but should work much better. And even if it should become "out of date", the only drawback will be that Visage sometimes will be a bit slower than needed. ;) Should hopefully remove the need for the **RTG** switch.

Made some more changes for improved RTG support.

Fixed a bug that could cause crashes when started from Workbench via a project icon.

1.102 Visage.Guide/Program history/Version 39.3

Version 39.3

Compile date: December 12, 1995 Program size: 95152 bytes

TEST option did not work properly. It said "Not enough memory" for all non-datatype pictures, and did not display picture information for datatype pictures. Note that it seems like most datatypes doesn't report any error, even if the picture did contain detectable errors.

Rewrote the scaling functions. The new ones have much higher precision (and are smaller and simpler, without loss of speed). The scaling should always make a perfect screen fit now! ;)

Made some other scaling related changes, so that Visage (hopefully) will make the picture completely visible at all times.

Visage did not properly parse the density (i.e. "reversed" aspect) information in JPEG pictures.

Tried to remove the trashed graphics that sometimes can be seen to the right of a picture, when displayed on a Picasso emulation screen.

Made some further optimizations on the dithering functions (cases DITHER NONE and DITHER ORDERED). Overall speed increase at most some 6%.

GIF pictures that were larger than specified (in the header) caused problems for Visage.

Fixed a bug in the **SORT** option, that could cause crashes and similar.

Added nifty **SHOWINFO** switch. Displays picture information above the actual picture.

Added **INFO** switch. Makes Visage display information about the picture, but not load it.

Cleaned up the datatype reader a bit. Did not improve behavior as I had hoped (found the reason for it after I had made the change ;).

Replaced DoIconLite with **UpdateTT**. I wrote UpdateTT since DoIconLite did not quite behave the way I wanted. UpdateTT is also much smaller. ;)

Tried to fix the crashes when promoting IFF ILBM pictures to CyberGraphX modes.

Improved the **TOFRONT** behavior a little.

French translation by Georges Goncalves included.

1.103 Visage.Guide/Program history/Version 39.4

Version 39.4

Compile date: December 22, 1995 Program size: 47692 bytes

Fixed a problem that trashed memory when displaying (non-scaled) interlaced GIFs.

Missing/incorrect password got no error message. I knew about the problem before I released version 39.3, but I forgot to fix it! ;)

The "Program size:" field in the version 39.3 information was with all debug info (it is quite much ;).

Special screen modes (HAM, EHB and similar) was not properly handled when a MODE argument was specified, possibly causing pictures to be displayed in the wrong special mode.

The **LOWMEM** option caused crashes if "show next picture" was requested while loading the current picture.

1.104 Visage.Guide/Program history/Version 39.5

Version 39.5

Compile date: January 21, 1996 Program size: 43832 bytes

Fixed a bug that caused Visage to not work on CyberGraphX screens (and quite likely other intuition emulators as well). Introduced in version 39.3 or 39.4.

Recompiled with SAS/C 6.56. Reduced size with almost 4 KB.

Fixed an Enforcer hit that occurred when starting Visage from a shell with the **QUIET** option.

Visage did not always close datatypes.library as needed, when viewing datatype pictures.

German translation by Lars Eilebrecht included (I got one from Christian Kemp first, but his translation was not quite complete ;).

Added some more error checking in the GIF unpacker.

When breaking the current picture to view the next, Visage behaved as if the **LOWMEM** switch had been specified.

Visage did not propagate break signals that were received during a fade. Now the fading code simply peeks on the received signals and does not process them (leaving that for later). This means that Visage will not react immediately to break signals during the fade, but to fix that would cost more work than I would like to spend right now. ;)

Argh! Catalog generation went haywire in version 39.3. ./ Why did not anyone notice this earlier? ;)

If **SHOWINFO** was specified, but there were no **vertical centering** or **scaling**, then Visage could calculate the wrong screen height, and would quite likely trash some graphics memory.

Visage would quit on certain errors (such as file did not exist).

Switched on invisible typing in the password requester.

Visage sometimes ignored "show next picture" requests.

Misc cleanups and optimizations. I hope I did not break anything fixed above. ;)

Changed so that **UpdateTT** only requires OS 2.04+. I made it require OS 3.0 for no real reason. ;) Added a proper version string. Recompiled with SAS/C.

1.105 Visage.Guide/Program history/Version 39.6

Version 39.6

Compile date: 23-Mar-96 Program size: 45204 bytes

Removed some debug output in a BestMode related function (forgot to do it in 39.5 ;).

Some errors in the GIF reader wasn't properly reported.

Certain "early" picture errors caused Visage to crash.

The IFF reader would crash when rendering to foreign screen modes (including when the **RTG** switch had been used, since it makes Visage assume all modes are foreign).

Interlaced GIF pictures weren't always rendered properly. It seems like it didn't always render the final pixel line. While at it, I cleaned up the responsible code.

Fixed a quirk in the Installer script (selecting English catalog for installation didn't work).

The color selection for the **SHOWINFO** text didn't always produce the correct result, possibly making the text invisible.

Added pattern support for the **MONITOR** argument.

Added the **NOICON** switch. Makes Visage ignore any `#?.info' files encountered.

Increased the date buffer, when displaying the about message. This to solve the problem for an Italian user, with a buggy Italian locale. Since I assume other users (mostly Italian, I guess ;) can have this problem as well (i.e. Visage crashes when right after displaying the about message), I made this simple workaround.

Changed the **NORMALNAMES** behaviour a little. When specified, then the display database names will be checked first, when using the **MODE** argument. The behaviour here was not clearly defined earlier.

Visage will now remap datatype pictures the system can't display. This is needed for proper behaviour on ECS Amigas. This forced me to change back to the old (and somewhat slower) planar to chunky converting code, as it is more generic. It is still much faster than the ROM code. ;)

Added the **NOFOREIGN** switch, to make Visage work better with certain Intuition emulators (most notably the PicassoII). If Visage tended to crash for you - especially when you didn't specify a display mode yourself - then you should try this switch.

Some minor internal improvements...

Improved argument handling for tooltypes. Strings with quotation characters were not properly handled. The main candidate for this problem was the **COMMAND** argument.

Improved mode selection for PCHG pictures (so that modes that do not support such pictures are avoided).

French and Italian catalogs updated.

1.106 Visage.Guide/Program history/Version 39.7

Version 39.7

Compile date: 12-Apr-96 Program size: 45168 bytes

Note: Version 39.6 was only available from my home page. For reasons unknown to me, I was unable to upload it to Aminet. I only got messages about corrupt archives, while the archive was ok when I tested it.

Workbench argument parsing didn't work properly. Sorry, I hardly ever start Visage from the Workbench! ;)

The bottom of interlaced GIFs still didn't look quite right. I think I've got it right now.

Errors that occurred during the internal IFF ILBM or GIF unpacking got the wrong error message.

1.107 Visage.Guide/Program history/Version 39.8

Version 39.8

Compile date: 96-05-06 Program size: 45232 bytes

Finally found and fixed the bug that caused crashes when the system ran low on chip memory.

Updated the installer scrip to really work when english language was selected during install (an updated 39.7 archive with this fix has been available on my home page).

Updated the installation script: Made changes for better V43 conformance, and added some other things.

Finally updated the Swedish translation. ;)

1.108 Visage.Guide/Program history/Version 39.9

Version 39.9

Compile date: 96-07-02 Program size: 45040 bytes

Visage should now work better with the V43 picture.datatype and truecolor (24-bit) images.

Datatype re-rendering was not correctly done for HAM8 pictures.

Made some other minor changes and cleanups to the datatype reader.

The text displayed by the **SHOWINFO** switch was not displayed on the right position, sometimes making text clipped (due to the font I normally use, I did not notice the problem. I know; should have tested with some other fonts ;).

The task creation code were doing somewhat naughty things, causing Mungwall hits (however, this did work with the current OS versions). Changed to more proper behaviour.

Russian and Spanish translations included.

Clarified the behaviour of the **VisageOpts** configuration.

Visage did not always clean up properly when aborted. This would usually only happen for datatype pictures.

Visage did not behave consistently if **WAITFORPIC** was specified, and you asked Visage to view the next picture. Now Visage will never wait in this particular case. Cleared up the manual on this part as well.

Tweaked the startup code a little.

Funny... I just found out that (most) DOS function indeed does support soft links, at least in OS 3.1. How come I did not notice this before, when I added the "soft link support" code in Visage in the first place? As I think of it, the error code was a bit odd. :) Anyway, that code have now been removed, saving a few bytes.

Upcoded **MakeLink** to C and added soft link detection. It will now state that links between volumes needs the SOFT keyword, rather than saying something like "object not of require type".

1.109 Visage.Guide/Program history/Version 39.10

Version 39.10

Compile date: 96-09-26 Program size: 45088 bytes

Increased minimal bitmap width from 16 to 80, when running under OS 3.0. This should fix the improperly displayed narrow pictures (i.e. those less than 80 pixels wide).

When specifying the **NOICON** switch, then the file requester will not include any icons either.

Visage now sanity checks some picture parameters to prevent problems. Earlier, pictures with e.g. a depth of zero would cause crashes.

Rendering deep grayscale (more than 16 shades) pictures on ECS Amigas with dithering turned off didn't look very well. Fixed.

Visage always (?) treated files selected in a file requester as having patterns, when MuFS was installed, due to a bug in Visage. As a result, files containg pattern characters couldn't be selected (without some extra work).

Added spanish translation of the installer script.

1.110 Visage.Guide/Program history/Version 39.11

Version 39.11

Compile date: 96-12-08 Program size: 45148 bytes

Norwegian translation updated. Added translations for Dutch (I've had the actual translation since 39.8, but no catalog was made for it, nor was the install script updated ;) and Serbian (at least I assume that is what Srpski stands for ;). Updated the installer script to install Russian catalogs as well.

Added the **NOFILECHECK** switch, and changed the behaviour in the file requester a bit. If the switch is not specified, then Visage will always check if a specified file exists, rather than process it as a pattern. This solves the problem with failing to display pictures with pattern characters in the name or the path, especially when Visage is started from the Workbench. The drawback is that patterns might not always be treated as you like. Hence the switch.

The release archive is now created a bit differently. Should hopefully fix the uploading problems I've experienced (most likely caused by Unix LhA failing on the Visage archive. I've removed some archiving options likely to cause problems according to Stefan Boberg). The 39.10 release didn't make it either... :)

Added the **BORDERBLANK** option.

Recompiled with SAS/C 6.57.

Wrote the **DeleteLink** command.

Added version string to **MakeLink** , and recompiled it with SAS/C 6.57.

1.111 Visage.Guide/Program history/Version 39.12

Version 39.12

Compile date: 97-04-27 Program size: 116776 bytes

Hey, this was originally meant to be a minor update! ;)

The **NOFILECHECK** option in 39.11 had the side effect of making Visage complain about lack of memory when started from the Workbench, and the **COMMAND** being executed in the wrong drawer sometimes.

Added the **PATTERN** argument, where a pattern for the file requester can be specified.

Increased the minimal screen width under OS 3.0 to 128 pixels.

Fixed a few quirks in the installer script.

PNG support added, using libpng and zlib. Note that libpng is on the large side... ;)

Added the **NOPNG** switch because of the PNG support.

Rewrote the JPEG loader, to use the IJG code directly rather than the jpeg.codec, in order to make Visage support progressive JPEG pictures. It also makes Visage a bit larger, but on the other hand, the total memory usage when displaying JPEGs will be lower (the jpeg.codec contains a lot of code not used by Visage). JPEG reading should also be somewhat faster.

Asyncio library release 9 used.

Removed bad graphics that could be seen on the right hand side of (some) very narrow truecolor pictures, when not displayed on a truecolor screen.

Soft links where not properly handled in the pattern matching loop.

Reworked the internal readers to be more pipe friendly, when possible. Only datatype pictures can not be read from a pipe now. Please note that even if the asynchronous IO buffers are not used by datatype pictures, they will be allocated anyway. Just a word of warning for you with large buffers (set via **BUFSIZE**). ;)

If xpkmaster.library version 3 or higher is used, then XPK-compressed (IFF ILBM) pictures are read using a custom IO hook. This should enable reading such pictures from a named pipe, as long as the file is not PowerPacked (Xpk does "limited" seeks for normal Xpk-files (but seeks to the end of PowerPacked files), so the buffers should be able to "catch" the seeks). Note also that this is somewhat experimental, as the IO hook part of Xpk is not that well documented yet.

Reduced the default value for **BUFSIZE** to 8 KB.

Specifying "Visage FileThatDoesNotExist FOREVER" would place Visage in an endless loop. Now you can break it (with CTRL-C).

Due to the Unisys LZW patent, and the conditions for using LZW for GIFs in freely distributable programs, the GIF support has been removed. The **NOGIF** argument is still in the template (for now) for backwards compatibility.

Improved the datatype reader, to check if the returned bitmap is displayable. If not, the bitmap is copied to a proper screen. This will increase the memory usage, but will ensure that the picture is displayed as intended.

CyberGraphX support added! Was rather easy actually. An hour work or two did the trick. Ok, the RTG support code did make things rather easy. ;)

Changed the **COLOR** switch to a keyword with various options, for the CyberGraphX support.

Improved support for picture.datatype version 43.

The first loaded picture was not always brought to the front. Mostly noticed for datatype pictures when displayed on a graphics card screen.

Algorithmic mode names are now used by default for CyberGraphX modes, and they look like this: "<Monitorname>:<Width>x<Height>". Similar to the normal names, but now you do not need to worry about the number of spaces in the string. ;)

String arguments in **VisageOpts** can now be reverted to the default by specifying "".

DRAWER argument added.

On ECS Amigas, Visage would sometimes fail to find a suitable mode for IFF ILBM pictures using extra halfbrite.

Seems like the reading of Xpk encrypted pictures has not been working for a while, due to unexpected behaviour in Xpk.

Changed file list handling code (responsible for the **RANDOM** and **SORT** arguments), to use less memory, and be faster during program exit (though possibly a bit slower during drawer scanning, at least if many drawers are scanned). Also changed to a sorting function posted by (and, I presume, written by) Wessel Dankers, in the news group comp.sys.amiga.programmer.

Plugged a few memory leaks.

Changed the layout of this manual quite a bit (as may have noticed by now). You need AmigaGuide version 40 or better for it to be displayed properly. As a result of these changes, the FixAG program is no longer needed, nor included.

Translations updated by ATO.

Changed random function, to something that hopefully is better. Not that the old one was bad really, but... ;)

1.112 Visage.guide/Index

Index

A **Acknowledgements** Acknowledgements **Algorithmic mode names** Algorithmic mode names **ALL** **ALL** **ARexx scripts** Included programs **Arguments** Arguments **Aspect, bad** Aspect **Author information** Author information

B **Back picture visible** Back picture visible **Background** Introduction **Bad aspect** Aspect **BESTMODE** **BESTMODE** **BORDERBLANK** **BORDERBLANK** **BUFSIZE** **BUFSIZE** **Bug reports** Author information **Bugs** Notes

C **CENTER** **CENTER** **CLIPBOARD** **CLIPBOARD** **COLOR** **COLOR** **Color cycling** **CYCLE** **COMMAND** **COMMAND** **COMPACT** **COMPACT** **ConfigPath** VisageOpts **CyberGraphX support** Rendered pictures **CYCLE** **CYCLE**

D **DATATYPES** **DATATYPES** **DELAY** **DELAY** **Delay background loading** **LATELOAD** **DeleteLink** **DeleteLink** **Disclaimer** No warranty **DITHER** **DITHER** **DRAWER** **DRAWER**

E **ECS** **ECS** **Environment variables** VisageOpts **Examples, mode name** Examples **Executing commands** **COMMAND**

F **FADE** **FADE** **Features** Features **FILES** **FILES** **Floyd-Steinberg dither** **DITHER** **FOREVER** **FOREVER** **Freeware** License **Future, the** The future

G **GAMMA** **GAMMA** **GetModeID** **GetModeID** **Graphic cards** Graphic cards **Graphics overscan** **OVERSCAN** **GRAY** **GRAY**

H **HAM** **HAM** **Highcolor render** **COLOR** **History** Program history **Horizontal screen centering** **CENTER**

I **I** Author information **Ignore mouse clicks** **NOCLICK** **Included programs** Included programs **INFO** **INFO** **Information, author** Author information **Information, legal** Legal information **Interlaced JPEG and PNG pictures** Interlaced JPEG and PNG pictures **Introduction** Introduction

J **Jumping mouse** Mouse jumping

K **Keys** Viewing keys **Kill** **Kill** **Known problems** Notes

L **LATELOAD** **LATELOAD** **Legal information** Legal information **Libraries, required** Required libraries **License** License **LOWMEM** **LOWMEM**

M **MakeLink** **MakeLink** **Max overscan** **OVERSCAN** **Me** Author information **Mode name examples** Examples **Mode name notes** Notes **Mode name parts** Parts **Mode names** Algorithmic mode names **MODE** **MODE** **MONITOR** **MONITOR** **Monitor promotion** **MONITOR** **Mouse buttons** Viewing keys **Mouse jumping** Mouse jumping

N **No algorithmic mode names** **NORMALNAMES** **No background loading** **LOWMEM** **No warranty** **No warranty** **NOAUTOSCROLL** **NOAUTOSCROLL** **NOBUSY** **NOBUSY** **NOCLICK** **NOCLICK** **NODATATYPES** **NODATATYPES** **Nodes, mode name** Notes **NOENV** **NOENV** **NOFILECHECK** **NOFILECHECK** **NOFLICKER** **NOFLICKER** **NOFOREIGN** **NOFOREIGN**

[NOGIF](#) [NOGIF](#) [NOICON](#) [NOICON](#) [NOIFF](#) [NOIFF](#) [NOJPEG](#) [NOJPEG](#) [NOPNG](#) [NOPNG](#) [NORMALNAMES](#) [NORMALNAMES](#) [Notes](#) [Notes](#) [NOWARN](#) [NOWARN](#)

O [Ordered dither](#) [DITHER](#) [OVERSCAN](#) [OVERSCAN](#)

P [Parts, mode name](#) [Parts](#) [PASSWORD](#) [PASSWORD](#) [PATTERN](#) [PATTERN](#) [Pictures, rendered](#) [Rendered pictures](#) [Pipe reading](#) [Pipe reading](#) [PNG support](#) [PNG support](#) [POINTER](#) [POINTER](#) [Priority](#) [TOOLPRI](#) [Problems](#) [Notes](#) [Program history](#) [Program history](#) [Programs](#) [Included programs](#)

Q [QUIET](#) [QUIET](#) [Quirks](#) [Notes](#)

R [RANDOM](#) [RANDOM](#) [Rendered pictures](#) [Rendered pictures](#) [Reporting bugs](#) [Author information](#) [Required libraries](#) [Required libraries](#) [Requirements](#) [Requirements](#) [Rexx scripts](#) [Included programs](#) [RTG](#) [RTG](#)

S [SCALE](#) [SCALE](#) [Screen centering](#) [Screen centering](#) [Screen centering, horizontal](#) [CENTER](#) [Screen centering, vertical](#) [VCENTER](#) [Screen gamma](#) [GAMMA](#) [Screen mode](#) [MODE](#) [Screen mode names, algorithmic](#) [Algorithmic mode names](#) [Screen overscan](#) [OVERSCAN](#) [Shell](#) [Arguments](#) [SHOWINFO](#) [SHOWINFO](#) [SORT](#) [SORT](#) [Standard overscan](#) [OVERSCAN](#) [System requirements](#) [Requirements](#)

T [TEST](#) [TEST](#) [Text overscan](#) [OVERSCAN](#) [Thanks to...](#) [Acknowledgements](#) [The future](#) [The future](#) [TIME](#) [TIME](#) [TIMES](#) [TIMES](#) [To do](#) [The future](#) [TOFRONT](#) [TOFRONT](#) [TOOLPRI](#) [TOOLPRI](#) [Tooltypes](#) [Arguments](#) [Tooltypes ignored](#) [Tooltypes ignored](#) [True-color render](#) [COLOR](#)

U [UNIT](#) [UNIT](#) [UnpackILBM](#) [UnpackILBM](#) [UpdateTT](#) [UpdateTT](#) [Usage](#) [Arguments](#)

V [VCENTER](#) [VCENTER](#) [Vertical screen centering](#) [VCENTER](#) [VERBOSE](#) [VERBOSE](#) [Version history](#) [Program history](#) [Video overscan](#) [OVERSCAN](#) [Viewing keys](#) [Viewing keys](#) [VisageOpts](#) [VisageOpts](#) [Visible back picture](#) [Back picture visible](#)

W [WAITFORPIC](#) [WAITFORPIC](#) [WBMONITOR](#) [WBMONITOR](#) [Why Visage?](#) [Introduction](#) [Workbench](#) [Arguments](#)