

## **SuperView-Library**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> SuperView-Library		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 25, 2024	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>SuperView-Library</b>	<b>1</b>
1.1	SuperView Library Documentation . . . . .	1
1.2	Copyright . . . . .	3
1.3	Disclaimer . . . . .	4
1.4	Distribution . . . . .	4
1.5	Abilities, Purposes and much more . . . . .	6
1.6	Making use of 680x0 CPUs . . . . .	6
1.7	Installation and Configuration . . . . .	7
1.8	Overview of currently available SVOjects . . . . .	10
1.9	Overview of currently available Device SVOjects . . . . .	11
1.10	Overview of currently available SVDrivers . . . . .	11
1.11	Overview of currently available SVOperators . . . . .	12
1.12	Software supporting SuperView-Library . . . . .	13
1.13	superviewnote . . . . .	13
1.14	sqopalnote . . . . .	14
1.15	superloadernote . . . . .	15
1.16	imageengineernote . . . . .	15
1.17	picmanagernote . . . . .	15
1.18	drafunote . . . . .	16
1.19	And thanks for all the fish: . . . . .	16
1.20	How to contact the author . . . . .	19
1.21	Official SuperView Online Support . . . . .	20
1.22	The future of SuperView-Library... . . . .	21
1.23	Known bugs and suggestions for workarounds . . . . .	21
1.24	Harddisk MaxTransfer Problem . . . . .	21
1.25	Problems with specific applications . . . . .	22
1.26	History . . . . .	23
1.27	Printer.svobject . . . . .	32
1.28	Scanner.svobject . . . . .	32
1.29	AmiFIG.svobject . . . . .	34

---

1.30 Degas.svobject . . . . .	35
1.31 ILBM.svobject . . . . .	36
1.32 PBM.svobject . . . . .	37
1.33 ACBM.svobject . . . . .	38
1.34 Datatypes support . . . . .	40
1.35 PCX.svobject . . . . .	41
1.36 SVG.svobject . . . . .	42
1.37 GIF.svobject . . . . .	44
1.38 GPlot.svobject . . . . .	46
1.39 BMP.svobject . . . . .	47
1.40 WinIcon.svobject . . . . .	49
1.41 FBM.svobject . . . . .	49
1.42 Limbo.svobject . . . . .	50
1.43 PNM.svobject . . . . .	53
1.44 PNG.svobject . . . . .	54
1.45 C64.svobject . . . . .	55
1.46 CDR.svobject . . . . .	56
1.47 IMG.svobject . . . . .	57
1.48 TIFF.svobject . . . . .	58
1.49 EPS.svobject . . . . .	59
1.50 GhostScript.svobject . . . . .	61
1.51 Targa.svobject . . . . .	64
1.52 WPG.svobject . . . . .	66
1.53 SunRaster.svobject . . . . .	67
1.54 SGI.svobject . . . . .	68
1.55 PICT.svobject . . . . .	69
1.56 Pictor.svobject . . . . .	70
1.57 MAC.svobject . . . . .	71
1.58 JPEG.svobject . . . . .	72
1.59 PCD.svobject . . . . .	74
1.60 FastILBM24.svobject . . . . .	75
1.61 YUVN.svobject . . . . .	76
1.62 DEEP.svobject . . . . .	78
1.63 QRT.svobject . . . . .	79
1.64 C-Source.svobject . . . . .	80
1.65 UtahRLE.svobject . . . . .	81
1.66 CyberDataType.svobject . . . . .	81
1.67 ECS.svdriver . . . . .	83
1.68 AGA.svdriver . . . . .	84

---

1.69 CyberGraphics.svdriver . . . . .	86
1.70 EGS7.svdriver . . . . .	87
1.71 PicassoII.svdriver . . . . .	89
1.72 OPAL.svdriver . . . . .	91
1.73 Retina.svdriver . . . . .	94
1.74 MERLIN.svdriver . . . . .	96
1.75 XOR.svoperator . . . . .	97
1.76 24BitToHAM.svoperator . . . . .	98
1.77 Crop.svoperator . . . . .	99
1.78 Dither24Bit.svoperator . . . . .	100
1.79 HilbertDither256.svoperator . . . . .	101
1.80 AnyTo24Bit.svoperator . . . . .	102
1.81 ExtractGrayScales . . . . .	103
1.82 ExtractRed . . . . .	104
1.83 ExtractGreen . . . . .	105
1.84 ExtractBlue . . . . .	106
1.85 TopToBottom . . . . .	106
1.86 LeftToRight . . . . .	107
1.87 Rotate . . . . .	108
1.88 RotateFree . . . . .	109
1.89 Scale50 . . . . .	110
1.90 CallPNM . . . . .	111
1.91 OptimizePalette . . . . .	112
1.92 PaletteDither.svoperator . . . . .	113
1.93 Requirements for the SuperView-Library Package . . . . .	114
1.94 NotesAndHints . . . . .	115
1.95 Memory Usage . . . . .	116
1.96 Displaying 24 bit graphics . . . . .	117
1.97 Converting 24 bit graphics . . . . .	117
1.98 SVPrefs . . . . .	118
1.99 SuperViewSupport-Library . . . . .	120
1.100SuperView in the Press . . . . .	122
1.101Books and other written stuff used during development . . . . .	124
1.102Other Program Projects . . . . .	125
1.103Credits . . . . .	126
1.104ControlPad Fileformat . . . . .	132
1.105CPInfo Fileformat . . . . .	134
1.106ControlPad Overview . . . . .	138

---

## Chapter 1

# SuperView-Library

### 1.1 SuperView Library Documentation

superview.library V15.4

- Freeware (Licenseware) -  
product-specific Licenseware

Any usage from and by other programs without an  
explicite license is strictly forbidden (see "Distribution").  
Ask for licenses.

© 1993-96 by Andreas R. Kleinert. All rights reserved.

A PerSuaTiVe SoftWorX PRODUCT.

This program has been written under OS V3.1 and is therefore  
fully compatible. It needs OS V2.04+.

Release Date: 28.9.1996

Legal  
Copyrights and legal stuff  
Disclaimer  
Distribution

Usage  
Short: Purpose and Abilities  
Requirements  
Installation

68020-060 support  
Making use of fast CPUs

Contact  
How to contact the author  
Online Support Services

Archive  
History  
SVObject Descriptions/History

---

Device SVOBJECT Descriptions/History  
SVDriver Descriptions/History  
SVOperator Descriptions/History  
Support-Library Descriptions/History

Various Topics

Supporting Software

Known bugs and workarounds  
Notes and Hints  
Possible future enhancements ...

Credits and even more legal stuff  
Thanks and Greetings  
SuperView in the Press  
Bibliography  
More Projects

ControlPad Preferences

Preferences

ControlPad Fileformat  
CPInfo Fileformat  
ControlPad enlistment

External Links in this Directory  
About Power-Brei (German Disk Magazine)  
German ReadMe File

\_ //  
Only \X/ Amiga makes it possible!

Please visit:

SuperView WWW Site  
[http://home.t-online.de/home/Andreas\\_Kleinert/main.htm](http://home.t-online.de/home/Andreas_Kleinert/main.htm)

Simon Edward's Image Engineer WWW Site  
<http://minyos.its.rmit.edu.au/~sbe/>  
(Image Processing program produced by Simon Edwards)

Die CHAOS-Theorie:

"Dabei geht es zum Beispiel darum, den verdammten Schmetterling zu finden, dessen Flügelschläge die vielen Stürme in letzter Zeit verursacht haben." (Terry Pratchett in "Total verhext")

In English means something like:

The Chaos theory for example deals with trying to catch the god damned butterfly, which caused the many storms the last times.

---

(see "Witches Abroad" by Terry Pratchett)

Ahm...well:

...and thanks for all the fish.

## 1.2 Copyright

The `superview.library` and the distributed files - e.g. the documentation files - are (C)opyright 1993-96 by Andreas R. Kleinert.  
All rights reserved.

(For some files there may additional or substitutive Copyrights take place, which then are stated locally within the documentation or via local reference to "Credits".)

The following usage and license conditions are announced for all parts of the distribution, which means `SVObjects`, `SVDrivers`, `SVOperators` and all other files.

When referring to the whole program package as such, it is called "SuperView-Library".

The usage and distribution of SuperView-Library takes place under the concept of product-specific Freeware (Licenseware).

For more details on distribution rules and developers restrictions, you have to see paragraph "Distribution".

Please note:

- \* Should any of the listed terms and clauses within this document not be valid in conjunction with the law of certain countries this does not affect the validity of the other clauses.
  - \* Some of the mentioned names or products within this or other documents may be copyrighted by companies or trademarks of companies or persons.
  - \* This software is based in part on the work of the Independent JPEG Group (concerning `JPEG.svobject`).
  - \* Some parts of this software do make use of the LZW (de-)compression algorithm, which is protected by patent claims in some countries (in Germany, algorithms as such are not protectable, especially when these have been published for the public).  
When using licensed versions of this library with commercial programs or on a for profit-basis, you may also have to license this algorithm (take a look at web page <http://www.unisys.com/>).  
The author of SuperView-Library does not feel responsible for offenses of third party people against that (concerning `GIF.svobject` and parts of `TIFF.svobject`).
-



If there are any legal doubts - YOU, as the user, must use the supplied and explained methods to disable GIF.svobject and parts of TIFF.svobject to be allowed using the library...

## 1.3 Disclaimer

The author takes no responsibility for any results of the use of this program.

This software is provided "AS IS" and there is no warranty of any kind, so that you use this software at your own risk.

The author reserves the right to discontinue development of the program.

## 1.4 Distribution

Distribution in Common

~~~~~

The Library must not be distributed isolated, detached from any specific program package, whose author has licensed the library.

If the licensed library is included within the licensing program's program distribution, the distribution conditions of this program take place as long as these do not explicitly contradict to the disposals described in the following text.

Licensed programs are not restricted to have any special legal copying conditions, so the Library may appear together with any form of Software, like e.g. Freeware, Shareware or commercial programs, perhaps with exception of "pure" Public Domain.

Restrictions

~~~~~

The program SuperView-Library in this version is product-specific Freeware (Licenseware), which at first means, that a single distribution of the Library is as well forbidden as an unauthorized distribution together with programs of non-licensees.

Also, the Library MUST not:

- be included into other program's distributions without my explicitly written permission
- be copied as a single package without being directly related to a specific program
- without a license be copied and also not be used directly or - via any tricks - indirectly

Licensing

~~~~~

There is no way of automatic licensing via "agree-to-the-above-terms". Any license has to be given in written, non-electronic form. Interim-licenses may be promised via phone, email, etc but

---

have to be confirmed in paper form.

The only one, who is authorized to write and send licenses or promise licensing is the author of SuperView-Library, which is Andreas R. Kleinert.

If you want to license the library, please send me the following information:

STATUS - what's the status of your planned or actually existing program ?

Main categories are Freeware, Shareware, Commercial.

WHO - what's your (or your companies') name, address and phone number, maybe also email adress ?

WHAT - what kind is your program of ?  
Not any programmer of any kind of program will get a licence. I am not going to support programs, which may concern my vital interests by being direct competitors of other programs where I am involved as a participator, depending on the success of the programs. [ ;-) ]  
On the other hand there are various ways to synchronize programming efforts and to come to a solution, which is acceptable for both sides.

PLANS - according to the type of your program you might also tell me something about your plans for future versions of the program as long as it does concern a fundamental change in the program's concept and function (competitor)

SAMPLE - would be nice to see a copy of the previous, beta or final version of the program

PAYMENT - are you gonna pay for the licence ?

Freeware authors will not have to pay anything, if they get a license. Shareware authors might. Commercial programs will perhaps.  
Tell me what you think, then we'll talk about it.  
Fees aren't expensive in my opinion.

After Licensing

~~~~~

Each license is specific to the licensee and his wishes.

Common to all licensees is the following:

- The Right to use the unmodified library within the program or program-group it has been licensed for
  - Access to not publicly documented functions
  - The Right to distribute the library together with your program
  - The Right to integrate the library into the concept of your program as long as no basic parts have to be changed
-

- The Obligation to respect the distribution conditions and to add a ReadMe file and documentation remark reflecting this to the program
- The Obligation to send me a copy of the final, registered or register-able version of the program using SuperView-Library

If there are more questions, please don't regret to ask me, because just asking doesn't cost you anything.

Developer Note:

Developer sources are available for C, Oberon2 and PCQ-Pascal, where E is planned and Asm is easy to derivate from the others (FD-files are available).

Not in all cases the newest versions are available (V9-V12; C is always up-to-date).

## 1.5 Abilities, Purposes and much more

The "SuperView-Library" consists of many functions and sub-libraries, which allow quick and easy displaying, saving, converting and processing of various picture formats, like IFF-ILBM, GIF, JPEG, PCX, Targa, TIFF, PNG, BMP, ... (more than 30 altogether).

It has been designed for the purpose to display and process any common type of bitmap graphics as fast and as comfortable as possible on almost any Amiga hardware configuration.

External Loader-/Saver-Libraries (SVObjects), Graphic Card Drivers (SVDrivers) and various Operators (SVOperators) allow easy but flexible configuration, usage and expansion.

Programming documentation is available for license-takers, only.

## 1.6 Making use of 680x0 CPUs

When installing superview.library, the Installer script will automatically install versions for your 68020+ CPU.

Since there's no difference between 68020, 68030, 68040 or 68060 integer versions (at least with SAS/C), you perhaps need not bother about doing any further optimizations.

But, on the other hand, if you do own an 68020/030 plus 68881/68882 math coprocessor, or an 68040 or 68060, you may wish to make use of these as well.

Usually, usage of Amiga OS' mathieeee-Libraries does automatically manage the coprocessor support, but for some reasons, these

libraries are not used with SuperView-Library modules:

- they can't be shared between processes
- they are not actually optimized for 68040/060 as with OS 3.1

Unfortunately, the used mathtrans.library does not support an FPU.

But there's a solution: it is strongly suggested, to use some of the available math-Library replacements from Aminet.

FMATH from Martin Berndt, for example does replace all of them:

- mathieeedoubbas.library
- mathieeedoubtrans.library
- mathieeesingtrans.library
- mathtrans.library

It is strongly recommended to install V40.6 or a newer version from Aminet (Aminet:util/libs/FMath406.LHA).

## 1.7 Installation and Configuration

=====

Installation Options: Overview

-----

- A. Automatized Installer-based Installation
  - 1. Installer procedure
- B. Customized Installation
  - 1. Main installation procedure
  - 2. Setting Up Standard Configuration
  - 3. Plugging in 680x0 modules
  - 4. Plugging in PackerSupport
  - 5. Plugging in SQ-Operators
  - 6. Using VMEM: (Needed)
  - 7. Plugging in Ghostscript/Postscript (TM)
- C. To Be Checked Also
  - 1. Datatypes-Support
  - 2. Other Libraries
  - 3. Additional settings
  - 4. Preferences
  - 5. LZW License Check

=====

A. Automatized Installer-based Installation

=====

- 1. Installer procedure

-----

Depending on the selected skill mode (novice, intermediate, expert) you will at least be asked for some of the following:

- your basic graphics engine (ECS, AGA, various GfxCards)
- a place to install the libraries to instead of LIBS:

- about LZW's legal status in your country
- a place to install the catalogs to instead of SYS:locale/catalogs
- a path to assign VMEM: to (s:user-startup will be modified)
- whether SVPrefs should be copied to SYS:Prefs
- whether to install the PackerSupport stuff
- whether to immediately also install the SQ-Operators

## B. Customized Installation

=====

### 1. Main installation procedure

-----

Copy "superview.library" and "superviewsupport.library" to your LIBS: directory, the SVObjects to "LIBS:svobjects/", the SVDDrivers to "LIBS:svdrivers/" and the SVOperators to "LIBS:svoperators/"!

Be aware, that library names are case-sensitive and wrongly spelled libraries won't be recognized, though.

Installation of the libraries can be done to any directory, so that these may be installed separate from the system libraries. Adding the SuperView-Library to the system would then be managed via

Assign LIBS: [PATH] ADD

somewhere in the User-Startup.

### 2. Setting Up Standard Configuration

-----

Docs/Sample\_Configs contains basic configuration settings for ENVARC: and ENV: which have to be copied there to allow the library system running without problems. env\_AGA contains plain AGA configuration and so on

- see Installer script for more information.

### 3. Plugging in 680x0 modules

-----

Owners of 68020-68060 systems should copy the turbo versions of some of the modules over the 68000 installation. Simply type "Copy libs/68030 to LIBS: all" or similar.

### 4. Plugging in PackerSupport

-----

PackerSupport/libs/svobjects does contain the support module for XPK and PowerPacker Data (PP20). Optionally copy this to LIBS:svobjects and the config from env to ENVARC: (and ENV: as well, of course)

### 5. Plugging in SQ-Operators

-----

Steve Quartly's great additional svoperators can be found within "SQ-Operators/". Just install those separately, as already done for PackerSupport before.

### 6. Using VMEM: (Needed)

-----  
 Additionally, you should create an assignment called "VMEM:", which is intended to contain virtual memory image files, which may be created either by superviewsupport.library (vmem\_XXXXXXXX), JPEG.svobject (jpeg\_XXX) or other modules.  
 It does not always make sense, to place this directory somewhere at a Ram-Disk, because most programs will perhaps only try to place files there, after they already checked, that there's no more memory available (so even your Ram-Disk would not grant more space).

It is suggested to place this Assign on a fast harddisk or partition, with a high AddBuffers value.

#### 7. Plugging in Ghostscript/Postscript (TM)

-----

If you have Ghostscript installed, you may want to use it for reading Postscript (TM) files - if not, you may consider to install it for such uses.  
 See GhostScript.svobject description for more information.

#### C. To Be Checked Also

=====

##### 1. Datatypes-Support

-----

"datatypes.library" V39+ is only needed, if OS3-DataTypes are wished to be supported - not available for OS2 users.

##### 2. Other Libraries

-----

Please take care, which libraries are also additionally needed by the single SVOjects/Drivers/SVOperators.

##### 3. Additional Settings

-----

###### Directory Caching:

~~~~~

Buffer values (set via C:AddBuffers) for the used Drives should at least be about 25, sometimes more may be quite useful.

###### Stacksize:

~~~~~

If you don't set this one to a reasonable value, the system might crash when using some specific modules (e.g. Dither24Bit).  
 It is strongly recommended to use values >= 32768 bytes.

##### 4. Preferences

-----

For modifying any specific ControlPad settings, you should use SVPrefs (optionally copied to SYS:Prefs during Installer-based installation).

Nevertheless sometimes modules do not come with the required .cpinfo files, so it would become necessary to edit ControlPad files directly at their locations ENV:SuperView-Library and ENVARC:SuperView-Library, maybe by using the descriptions inside

---

the doc-file "ControlPads".

#### 5. LZW License Check

If LZW usage is license-free in your country type the following two commands in your Shell:

```
SetEnv ENV:SuperView-Library/LEGAL.controlpad LZW=ENABLED
SetEnv ENVARC:SuperView-Library/LEGAL.controlpad LZW=ENABLED
```

otherwise

```
SetEnv ENV:SuperView-Library/LEGAL.controlpad LZW=DISABLED
SetEnv ENVARC:SuperView-Library/LEGAL.controlpad LZW=DISABLED
```

If you don't correctly answer this, you hereby agree in paying any possible license fees by yourself.

You may compare this with a freely distributable LZW source code that you decide to compile and use - by respecting the copyrights on it.

## 1.8 Overview of currently available SVOjects

Available SVOjects (mostly included) :

SVOject	Type	Read-Support	Write-Support
BMP (Win,OS/2)/RIFF	EXT	max 8/24bit files	max 8/24bit files
C64 (Koala, Doodle)	EXT	max 4bit files	(not yet available)
CDR (Corel Preview)	EXT	(various)	(not available)
CGM *	EXT	-> GPlot	(not available)
Atari Degas	EXT	max 4bit files	(not available)
EPS(F)-Header	EXT	(various)	only from 24bit sources
FastILBM24	EXT	24bit files	(-> via IFF-ILBM)
FBM (*nix)	EXT	max 8/24bit files	max 8/24bit files
FIG *	EXT	-> AmiFIG	(not available)
GIF87a/89a	EXT	max 8bit files	max 8bit files
Icon (W*nd*ws)	EXT	16 Color Icons	(not available)
IFF-ACBM	EXT	max 8bit files	max 8bit files
IFF-DEEP	EXT	24bit files	only 24bit files
IFF-ILBM	EXT	max 8/24bit files	max 8/24bit files
IFF-PBM	EXT	8bit files	8bit files
IFF-YUVN (misc)	EXT	24bit files	only 24bit files
IMG (GEM/V*nt*ra)	EXT	max 8bit files	(not yet available)
JPEG (JFIF)	EXT	max 8/24bit files	always as 24bit files
Limbo *	EXT	-> Limbo	(24 Bit via Limbo)
MAC (MacPaint)	EXT	max 1bit files	(not yet available)
PCD (PhotoCD)	EXT	24bit files	(not available)
PCX upto V3.0	EXT	max 8/24bit files	max 8/24bit files
PICT-2 (Mac)	EXT	always as 24bit files	(not available)
Pictor/PCPaint/PIC	EXT	max 8bit files	(not yet available)
PNG (Network Gfx)	EXT	max 8/24bit files	(not yet available)
PNM (*nix)	EXT	max 8/24bit files	max 8/24bit files
Postscript/PDF *	EXT	-> Ghostscript	(not available)

QRT/POV	EXT	24bit files	24bit files
SGI	EXT	max 8/24bit files	only 24bit files
SunRaster (RAS)	EXT	max 8/24bit files	max 8/24bit files
SVG Graphics	EXT	max 8/24bit files	max 8/24bit files
Targa	EXT	max 8-24bit files	max 8/24bit files
TIFF (V5.0)	EXT	max 8/24bit files	max 8/24bit files
Utah RLE	EXT	24bit files	24bit files
WPG (WP Draw BitMap)	EXT	max 8bit files	(not yet available)
OS3-DataTypes	INT	all Pictures/DTs	(not available)
picture.datatype V43	EXT	all Pictures/DTs	(not available)

\* SVOobjects marked with an Asterisk will only work, when the mentioned external programs (usually freely distributable software from AmiNet) also have been installed and set up as needed. (If not set up, these may be disabled via controlpad STATUS=DISABLED.)

There's also an Unpack.svobject in the PackerSupport-Directory, which allows reading of packed files without explicitly unpacking these before (supports XPK and PP20).

It is as well possible to save graphics as C-Sourcecode.

Planned SVOobjects (no guarantee !) :

- WMF/Metafile
- MTV
- Imagine Textures  
(if possible)
- RGB8/RGB24
- more Atari formats  
(if I get my hands on some pictures -> send some, if PD !)
- and more  
(you may send me gfx's and/or descriptions, if PD)

Do not send any graphics via email.

## 1.9 Overview of currently available Device SVOobjects

Available SVOobjects :

SVOobject	Type	
Printer	supports printer.device	EXPORT
Scanner	supports scanner.device of ScanQuix	IMPORT

## 1.10 Overview of currently available SVDivers

Available SVDivers (mostly included) :

SVDriver	Requirements	BITPLANE	ONEPLANE
----------	--------------	----------	----------

---



ECS	ECS*, OS V2.04+ (V37)	(system)	8/24bit
AGA	AGA*, OS V3.x+ (V39)	(system)	8/24bit
CyberGraphX	CyberGraphX RTG System	8bit	8/24bit
EGS-System	EGS-Graphic-Libraries V7	8bit	8/24bit
Merlin	Merlin Graphics Card	8bit	8/24bit
OpalVision	OpalVision Card	8bit	8/24bit
Picasso II	PicassoII-Card	8bit	8/24bit
Retina	Retina Z2/Z3 Card	8bit	8/24bit

\* ECS- and AGA.svdriver will both work with Graphic Cards, if they are configured with "BITMAPCOPY=RTG" and the GfxCard software does supply a Workbench Emulation and/or ScreenPromoter. If you own a Graphics Card, without having an appropriate SVDriver for it, always try using AGA.svdriver at first.

Planned SVDrivers:

"Send me a card, I write the Driver !"  
(at least trying to do so, then... ;-)

## 1.11 Overview of currently available SVOperators

Available SVOperators (mostly included) :

SVOperator

24BitToHAM	dithers 24 bit RGB to HAM6/HAM8
AnyTo24Bit	converts input to 24 bit
CallPNM	Call preset external PNM operator
Crop	"crops" boxes of any size from 8/24 bit graphics
Dither24Bit	dithers 24 bit RGB to 2..256 Colors
ExtractBlue	extracts Blue values from (upto) 256 Colors or 24 bit
ExtractGrayScales	converts (upto) 256 Colors or 24 bit to Gray
ExtractGreen	extracts Green values from (upto) 256 Colors or 24 bit
ExtractRed	extracts Red values from (upto) 256 Colors or 24 bit
HilbertDither256	dithers to Black & White
LeftToRight	turns left to right ...
OptimizePalette	reduces depth by filtering unused or redundant colors
PaletteDither	dithers (optionally in parts) to a user-supplied palette
Rotate	rotates by 90/180/270 degrees ...
RotateFree	rotates freely by 0..360 degrees about a given point
Scale50	scales to half/double size
TopToBottom	turns top to bottom ...
XOR	nice effects ...

Just for information:

- this distribution also includes a bunch of image processing operators by Steve Quartly (SQOperators) like

SQBentley.svoperator	SQBlur.svoperator
SQContrast.svoperator	SQDeep_Press.svoperator

SQDiffuse.svoperator	SQEmboss.svoperator
SQHighPass.svoperator	SQMosaic.svoperator
SQMotion_Blur.svoperator	SQOilPaint.svoperator
SQSharpen.svoperator	SQSlicing.svoperator
SQThreshold.svoperator	SQTiling.svoperator

- buyers of the commercial program Picture Manager professional (short: PMPro) will also get some additional operators, which are not freely distributable (exclusively shipped with PMPro):

Antique.svoperator	Brightness.svoperator
Complement.svoperator	Contrast.svoperator
Convolve.svoperator	DitherFix24.svoperator
EdgeDetect.svoperator	Gamma.svoperator
Mosaic.svoperator	RGBAdjust.svoperator
ScaleFree.svoperator	

(referring to Release V3.01 - may differ from version to version)

Planned SVOperators (no guarantee !):

- some more operators for common image processing
- and more

## 1.12 Software supporting SuperView-Library

- o The following programs do own a license for SuperView-Library:

Program	Description	Status
SuperView	THE image viewer and converter for anyone	SW
SqOpal	THE image viewer and processor for OpalVision	SW
SuperLoader	THE loader module for OpalPaint	FW
ImageEngineer	THE image processing program for anyone	SW
PictureManager	THE image database program for professionals	COM
DRAFU	THE function plotting program	SW

- o The following programs have been written by me and do also support SuperView-Library in some way :

Program	Description, Author	Status
SimpleView	SuperView-Library Example Program	FW
MicroView	SuperView-Library Example Program	FW
KFracPlus	fractal generator (saves via library)	FW
SIP	for examinations of SV-Modules	FW

## 1.13 superviewnote

SuperView is a program that has been designed for the purpose, to display any kind of graphic as fast and as comfortable as possible.

Fast means not only fast in reading and displaying but also in calling and using the program.

Highest possible flexibility is implemented via the external Driver-System, consisting of the SuperView-Library, the SVOobjects, SVDdrivers and SVOoperators.

These intentions resulted in a bundle of features and options you have access to when using SuperView and installing it to your System:

- licensed "SuperView-Library" (see directory SuperViewLibrary)
- intensive use of many special OS V2.04+ and OS V2.1+ capabilities
- support of many OS V3.x+ and AGA graphics features, as e.g. support of interleaved BitMaps
- support of all SVOobjects, SVDdrivers and SVOoperators of SuperView-Library
- "Screen-Grabbing"
- Commodity (optional)
- ARexx-Ports (optional)
- AppIcon (optional)
- AppMenu (optional)
- AppWindow (optional)
- Clipboard reading and writing
- Support of Devices, like e.g. Scanners
- Localization for OS V2.1+ (not all texts yet)
- AmigaGuide OnLine-Help for OS V2.04+ (asynchroneous & localized with 2.1+)
- detailed configuration via Config-File, Cli-Options and Workbench-Tooltypes
- Graphical User Interface (GUI) for Workbench-Users (optional)
- conversion of the supported File-Formats
- information about the displayed graphics via Requester, selectable via GUI-Menu
- and more

It is Shareware and can be found on AmiNet under gfx/show (for example gfx/show/SView563.LHA).

## 1.14 sqopalnote

SqOpal is an Image Processing, display and manipulation package especially for the OpalVision 24 bit Graphics and FrameBuffer hardware.

It has been written by Steve Quartly and Paul Huxham and optionally uses SuperView-Library.

It is Shareware and can be found on AmiNet under gfx/show (for example gfx/show/SqOpal20.LHA).

Steve Quartly: [steveq@sndcrft.DIALix.oz.au](mailto:steveq@sndcrft.DIALix.oz.au)  
Paul Huxham: [paulh@perth.DIALix.oz.au](mailto:paulh@perth.DIALix.oz.au)

## 1.15 superloadernote

SuperLoader is a Loader Module for OpalPaint (OpalPaint comes with the OpalVision 24 bit Graphics and FrameBuffer hardware).

It has been written by Steve Quartly and Paul Huxham and uses SuperView-Library.

It is Freeware and can be found on AmiNet in the graphics section.

Steve Quartly: [steveq@sndcrft.DIALix.oz.au](mailto:steveq@sndcrft.DIALix.oz.au)  
Paul Huxham: [paulh@perth.DIALix.oz.au](mailto:paulh@perth.DIALix.oz.au)

## 1.16 imageengineernote

Image Engineer is a shareware image processing application for any Amiga with 68020 and OS 2.x or greater.

Image Engineer can be used for tasks varying from converting images between different file formats, rendering 24 bit images down to standard Amiga screen modes, enhancing badly scanned images, applying special effects even up to advanced image composition.

What you can use it for is basically limited by what you can think of.

Registration is only 35 US dollars (45 AUS Dollars, 55 DEM).  
Registration sites are in Australia (Simon Edwards)  
and Germany (Andreas R. Kleinert).

Registered users will be sent a personal keyfile which will unlock all of Image Engineer removing the limits on the image size, and the 'Register Now...' requesters.

This keyfile will also work for future versions.  
Keyfiles can be sent out via mail or Email.

IE can be found on AmiNet under gfx/edit  
(for example gfx/edit/ImEngV3.1.LHA).

Simon Edwards: [sbe@yallara.cs.rmit.edu.au](mailto:sbe@yallara.cs.rmit.edu.au)  
IE WWW page: <http://minyos.its.rmit.edu.au/~sbe/>

## 1.17 picmanagernote

Picture Manager Professional 3.x (PMPro) is an excellent commercial image catalogization program with a large number of features for creation and handling of thumbnail tables.

It is a powerful database and allows loading, processing and displaying images via SuperView-Library and includes an automatic conversion function for catalogs of images, which allows to convert between all file formats SuperView-Library does support ... AND MORE ...

---

- The GERMAN version of Jürgen Schäfer's Picture Manager can be obtained from

IrseeSoft SPCS  
Meinrad-Spieß-Platz 2  
D-87660 Irsee                      Voice: +49-(0)8341-74327  
Germany                              Fax:     +49-(0)8341-12042

Or from any distributor. The official pricing as of May 1996  
has been 129 DEM (no guarantee).  
Ask them for it.

## 1.18 drafunote

DRAFU plus is a function plotter, which optionally supports  
SuperView-Library for saving the drawings from single windows.

It includes the following features:

- freely defineable function terms ( $x^?$ , sin, cos, tan, ...)  
(function parser by Dipl.Ing. Ulrich Degens)
- unlimited number of sizeable windows on the screen
- unlimited number of functions per window  
(presented as a list)
- upto 256 colors on the screen (AGA):
  - freely selectable draw color for each function
  - freely selectable background, drawing and axis color  
for each window
- loading and saving of function lists
- flexible axis' description (decimal, logarithmically, scientific)
- single/double axis description
- saving windows drawings in any of the graphics file formats  
SuperView-Library does support (optionally, library is not needed)
- hardcopy function

AND MUCH MORE

DRAFU can be found on AmiNet under misc/math  
(for example misc/math/DRAFU105.LHA).

Author: me ;-)

## 1.19 And thanks for all the fish:

I perhaps have to thank many persons for supporting me with ideas,  
Bug-reports and so on :

Thanks go to (in alphabetical order) :

\* Jan van den Baard

... for his great tool GadToolsBox, which I formerly used to

---

design the GUI of SVPrefs.

\* Ian O'Connor

... for the Designer, which I now do use for GUI creation.

\* Simon "ImageEngineer" Edwards

... for a lot of things concerning SuperView-Library :-)

\* Gerd Frank

... for Beta-Testing, Bug-Reports and for his many ideas and suggestions, concerning SuperView-Library and SuperView (also see notes there) !  
And last not least for the first draft of the Guide Documentation... ;-)

\* Roman Patzner

... for the nice icons he designed for use with Martin Huttenloher's MagicWB (8 Colors minimum) and sent to me for inclusion with SuperView. With SuperView-Library only the InstallerScript-Icon is used yet (several times ;-)

\* Steve Quartly and Paul Huxham

... for the various versions of OPAL.svdriver included with the library (Steve and Paul) and for Steve's SQ-Operators, included with his friendly permission.  
Steve also did "SuperLoader" and "SqOpal" together with Paul Huxham.  
THANKS !

\* Jürgen Schäfer

... for specific Beta-Testing and related Bug-Reports, as well as several useful hints and suggestions on implementing new features to the library.

\* Martin Schulze

... for uploading SuperView onto the AmiNet and including it into the SaarAG series, so that it reached more people out there.  
Also for sending and receiving all those mails, which did not find their way through the labyrinth of Fido-Gateways ;-)

\* Detlef Winkler

... for the new Icons for Doc- and AmigaGuide-files, included with the Library. He also designed some graphics and icons for use with SuperView and new drawer icons.  
Additionally, he had a lot of ideas, suggestions and graphics for 24 bit IFF-ILBM, IFF-YUVN, IFF-DEEP and SGI support.

\* to the translators

Italiano: Alessandro Basso (cralex@amiga.dei.unipd.it)  
          Luca Giolo (grifon@vega.unive.it)  
Svenska: Patrik M Nydensten (nyden@algonet.se)

---

Torbjörn Aronsson (torbjorn.aronsson@mailbox.swipnet.se)

\* and last NOT LEAST

- all \_registered\_ users of SuperView for supporting Shareware !!

\*\*\* COMMERCIAL BREAK - BEGIN \*\*\*

ORDER YOUR KEYFILE NOW !

\*\*\* COMMERCIAL BREAK - END \*\*\*

- the people mentioned below (still in alphabetical order ;-):

Torbjörn	Aronsson	for suggestions and bug reports
Alex	Carbin	TuC / Co-Sysop Century
Rüdiger	Dombrowski	ADX-Datentechnik
Thomas	Dorn	Author of XiPaint (thx for 3.2 and 4.0)
Sven	Drieling	"Indy" - Creator of "Power-Brei" DiskMag and the Oberon2-Language interfaces and modules.
Oliver	Eichhorn	author of EGSTasy
Thomas	Eigentler	Programmer of MERLIN.svdriver (included)
Thomas	Fischbach	did various online support for a long time
Fred	Fish	AmigaLibDisks and Fish CD-ROMs
Stefan	Grad	GPD-Disks
Klaus	Holtorf	for detailed information on graphic file formats
Ing.-büro	Helfrich	for supplying the PiccoloSD64 Graphics Card
Stefan	Kremer	TuC / Sysop Century
Alex	Lange	Time PD-Disks (former aps-electronic)
H.P.	Lattka	Franz PD-Disks
Andreas	Manewaldt	Taifun PD-Disks
Axel	Melzener	Game Object Design
Andreas	Neumann	Creator of the PCQPascal-Language includes and modules
Patrik M	Nydensten	for lots of suggestions and bug reports
Michael	Petrikowski	Amiga Szene PD-Disks, SEP - Software-Entwicklung
Albi	Rebmann	supplier of my ftn.sub.org UseNet domain/address
Frank	Taha	thanks for the DP II (PC) PBM graphics
Michael	Trautes	Sysop of Micky's box - my current Fido Boss
Michael	Trautmann	PC-Programmer, for information on Win icons
Christian	Wincziers	Sent me some PICT-2 graphics. Thanks !
Florian	Zeiler	Sending me the RetinaZ3 and PicassoII graphic cards enabled me to write own Drivers for these.

- some users, which e.g. reported bugs via mail, e-mail or telephone  
or did something else related to my programming work on the library  
(only the ones, which have not already been listed above) :

Thomas	Alexnat	
Joachim	Baumeister	
Rudi	Brand	
Aaron	Digulla	
Michael	Flad	
Grant	Fribbens	
Thomas	Gundlach	
Richard	Hartmann	
Mats	Jansson	(for reporting 060 problems)

Michael	Kilimann	(for reporting 060 problems)
Mika	Lundell	MERLIN testing and so on. Thanks.
Mats	Jansson	
Jim & Becky	Maciorowski	(thanks for the nice card :-)
William	Maddock	
Neil	Mohr	
Patrik	Nydensten	lots (tons ;-) of suggestions
Patrick	Ohly	
Andy	Philpotts	
Fabio	Rotondo	
Jürgen	Schneider	
Tommy	Simonsen	
Klaus	Stengel	
Reinhard	Theling	
Henrik	Tikanvaara	
Marco	Vernaglione	
Fabien	Wernly	
Tilo	Winkler	(thanks for the strange TIFFs ;-)

and more.

Maybe I forgot somebody to list here, but nobody's perfect... ;-)

Please note: If you wrote to me and didn't get an answer,  
this need not necessarily mean, that I didn't  
answer - mails can get lost sometimes.  
Please tell me and ask again!

## 1.20 How to contact the author

```

| You may reach me the following way.
| Send bug-reports, money or whatever to:
|-----
| * SuperView Development & Registration *
| * DRAFU Development & Registration *
| * Image Engineer Registration Site Europe *
|
|
| PerSuaTiVe SoftWorX
|
| Andreas R. Kleinert
| Sandstrasse 1
| D-57072 Siegen
| Germany, Europe
|
| Any snail mail to the old address will still be routed.
|
| Phone: +49-271-22869 also FAX + AM
|        +49-271-22838
|
| Weekdays after 17.00h.
|
| When calling via phone you may leave a message,

```



```
|         if I'm not available - but don't expect me      |
|         calling back to USA, Australia, ... since      |
|         german phone rates are HIGHLY expensive.      |
|_____|
```

[http://home.t-online.de/home/Andreas\\_Kleinert/main.htm](http://home.t-online.de/home/Andreas_Kleinert/main.htm)

eMail:

DO not SEND ANY binaries (or uuencoded) VIA THE  
FOLLOWING EMAIL ADDRESSES, EXCEPT MAYBE small ONES  
VIA t-online.de (smaller or equal 16 KB).  
THANK YOU.

```
- Fido      Andreas_Kleinert 2:2457/350.18
- Usenet
      Andreas_Kleinert@superview.ftn.sub.org  (Fido-Gate)
      Andreas_Kleinert@t-online.de           (T-Online)
      ARK@COB.wwbnet.de                     (Z-Netz)
```

- If nothing else works, try one of these public  
Fido-Usenet gateways:

In Germany:  
Andreas\_Kleinert@pl8.f350.n2457.z2.fido.sub.org

From USA or elsewhere:  
Andreas\_Kleinert@pl8.f350.n2457.z2.fidonet.org

Please note, that the "superview.ftn.sub.org"  
domain will perhaps be renamed in late 1996  
(it is BTW quite \_slow\_).

When reporting any bugs, please don't forget to include a detailed  
description of the bug and tell me, if it is reproduceable or not.  
Also mention the version number of SuperView-Library (and e.g. SuperView)  
which caused the bug and describe your system configuration  
(Amiga model CDTV/CD-32/600 (HD)/500 (+)/1000/1500/1200 (+/HD)/2000/2200/  
2500 (UX/AT)/3000 (+/T)/4000 (/030/040/060/PPC) (T)/DraCo (/040/060),  
Kickstart/OS Version, RAM, HardDisk, GfxCard, any special  
Hardware/Software)

Since there recently have been some problems: please don't forget  
to mention, whether your machine is accelerated in any way, or  
not (68020/.../060 plus 68881/882, PMMU or PowerUp/PPC).

## 1.21 Official SuperView Online Support

```
=====
Official SuperView Online Support Sites
=====
```

\*\*\* COB \*\*\*

- Running on AMIGA 4000 TOWER / 030 25Mhz / 10MB / 3.7 GB HD -

Sysop: Jens Loreck  
Online: 00.00h - 24.00h  
Data: +49-2774-92064 V34/ISDN (33.6k/64k bps)  
+49-2774-92065 V32terbo/ISDN (19.2k/64k bps)  
UseNet: SYSOP@COB.wwbnet.de

Guest-Account for downloads (user "GAST", no password).

This Box is AmBos-based, you may easily browse through the BBS areas.  
SuperView is stored at "Box Inhalt/Support/SuperView".  
(For AmBos, use ANSI/VT220 and 16 Colors.)

## 1.22 The future of SuperView-Library...

The Future (or: The Undiscovered Zone ;-)  
=====

Well, ideas cannot be planned and creativity is not able  
to be calculated ;-)

So I can only express, what I'm thinking about the future of the library:  
There are many things, which I want to implement, on the other hand  
there are things, which are just necessary to be implemented sooner  
or later. So much work has to be done (still).

I'll try to do this, but I hope that there's enough support from the  
Amiga programmers all over the world (via Inter/AmiNet, Fido, etc.)  
to help me to continue doing so.

This does not only mean financial support, but also additional  
programming efforts like third-party improvements on the library.

You may contact me for developer support - like already mentioned before.

- Andy

## 1.23 Known bugs and suggestions for workarounds

Harddisk MaxTransfer problem (workaround)  
Problems with specific applications (workaround)

Crashes on 060 systems should no longer appear - if the library crashes  
on specific modules at startup-time, please trace it back with SnoopDOS,  
delete that specific modules, and contact me for a bug-report.

## 1.24 Harddisk MaxTransfer Problem

---

### IDE-MaxTransfer Problem

-----

Since SuperView-Library usually holds very large buffers within memory, it also likes to read and write these completely from and to disk. This means, that the specific device drivers are confronted with quite large values of bytes to be read or written, which perhaps usually does not happen very often.

Sometimes the firmware of IDE-Harddrives, like shipped with the A4000/030-040 or A1200HD, does not support transfers of blocks larger than 64K (65535 Bytes) during one single write operation. Ususally the DOS splits larger writing calls to take care of this restriction. But since this is just a lack of performance and actually does not comply to the IDE/AT standard, the default value for this "MaxTransfer" is not 0xFFFF (64K) but 0xFFFFFFFF or 0xFFFFFFFF instead.

If any written graphics files are mysteriously damaged or will be read incorrectly (writing is more critical than reading), you should start your "HDToolBox" and select "Partition Drive" for the concerned HardDrive. After that activate "Advanced Options" and chose "Change". Modify the "MaxTransfer" field, so that it does reflect "0xFFFF" then. After that leave all the windows by confirming "OK" and select "Save Changes to Drive" (no longer disabled) on the first window.

```
>>> Do not change any other settings within "Partition Drive", if <<<
>>> you don't know, what you're doing, since actually partitioning <<<
>>> your HardDisk would cause your complete data to be lost. <<<
>>> If you did change something you didn't want to change, just <<<
>>> "Cancel" the whole thing and start from the beginning. <<<
```

Perhaps you'd like to know, why I did mention this here ?!

Well, some weeks after I bought a new M\*x\*t\*r 540 MB HD, SuperView-Library did seem to have x-time more bugs than before, which almost all could not be explained rationally: writing a "clean" buffer to disk in several file formats (did not matter), with this buffer still being valid after the write operation, resulted in damaged graphics after loading. If uncompressed, the data still was all there, but like a kind of mosaic, with always some blocks at the wrong places...

It took me some weeks to actually realize the bug itself and approximately two days to find out, \*why\* it happened... %-(

## 1.25 Problems with specific applications

Here's a list of several programs, which either cause SuperView-Library to crash, or which may crash or fail unexpected (so that it might seem, as if SuperView-Library did crash or were the reason of the fault):

- o EGS libraries V6 and V7 © VIONA Development

=====

### Problem

-----

When flushing the EGS libraries (at least after using the Amiga emulation mode), it seems that the libraries will cause recoverable

alerts with OS 3.x. Maybe on some systems crashes will occur.  
Don't know, whether the libraries are really the source,  
but it's likely.

Solution

-----

Don't flush ;-)

- o LX/020 1.03 © 1993 Xenomiga Technologie (Jonathan Forbes)

=====

Problem

-----

Obviously can't correctly extract large LZH/LHA archives.

- o NewMode V3.3 (and below) © 1992-95 by Andreas Linnemann

=====

Problem

-----

Has been reported to cause serious problems when running together  
with e.g. SuperView (when attaching a fixed ViewMode to the program).

Solution

-----

Already fixed for newer versions.

I, personally, would strongly recommend not to promote any SuperView  
Screens via NewMode, since now for almost any configuration there  
should be suitable SVDivers - alternatively also a RTG-configuration  
for AGA.svdriver.

- o VMM (supposed to be fixed)

===

Former versions of SuperView-Library did allocate all buffers  
with the MEMF\_PUBLIC flag set, which prevented e.g. VMM  
from storing these as virtual memory on disk.

As with current versions, the large image buffers - where  
it makes sense - are no longer prevented from being accessing  
by VMM. MEMF\_PUBLIC no longer is set for these.

Before - and with old modules - this could have been by-passed  
by setting the 10240 flag value within VMM's advanced options  
- but this should not be done any longer, since it may become  
dangerous under certain circumstances.

## 1.26 History

Please note the version-dependencies :

superview.library	SVObjects	SVDivers	SVOperators
Version 14-15	Version 4	Version 2-3	Version 3

Current version of superviewsupport.library: 7.2

V15.4 (28.9.1996) :

-----

- SVOobjects: - JPEG: upgraded to V6a (although not really useful)
  - Targa: fixed file recognition
- MISC: - version had not been bumped
- Docs: - V15.3 had been released 26 Sep 96, not 29 Sep 96 %-)

V15.3 (26.9.1996) :

-----

- MISC: slight changes in company name :->

V15.2 (19.9.1996) :

-----

- MISC: - .cpinfo file of ECS and AGA now uses \_\_SVOPERATOR tag instead of plain string (now opens Filerequester pointing to LIBS:svoperators)
- SVOobjects: - Targa: (see there)
  - GIF: writing transparent GIF89a files did not work due to a \*stupid\* bug
  - CyberDataType: was reported to cause crashes (see there)
  - ILBM, ACBM: (see there)
  - Scanner: (see there)
  - Limbo: minor fix (see there)
- SVDdrivers: - MERLIN: (see there; still slightly buggy)
- SVOperators: - Dither24Bit: added new dither mode: BURKES
- support-lib: - (see there)
- LIBRARY: - rewrote Installerscript for Installer V42.3:
  - behaves as usual with old installer
  - now is localized for new Installer
  - fixed some texts
  - tries to handle 68060 now
- caching SysBase now (SAS/C's \_\_USE\_SYSBASE)
- again revised startup-code (slightly)
- Docs: - removed 060 bug section
- rewrote

V15.1 (29.8.1996) :

-----

- NEW SVOobjects: - Printer: added DEVICE SVOobject for common Printer support
    - Degas: (see there)
  - SVOobjects: - ACBM: (see there)
    - TIFF: fixed serious bug for most <= 256 palette-color TIFFs
    - BMP: fixed bug in writing BMP header (see there)
    - DEEP: now supports reading 32 Bit RGBA
  - SVDdrivers: - MERLIN: (see there)
  - SVOperators: - Dither24Bit, PaletteDither: - improved
    - added BestPen mode
-

- new F/S dithering
- Dither24Bit: added new dither mode: ORDERED
- 24BitToHAM: - reworked
- Documentation:
  - rewrote "known bugs" section
  - rewrote "online support sites" section
  - updated note referring to VMM usage
  - fixed a bunch of spelling mistakes and typos ;-)
- Note:
  - sorry, for V14.1 to 14.3 - I simply should have done better testing on V14.1, but getting those nasty Enforcer hits fixed just made me euphoric and let me release SuperView V5.60 too soon ;-)
- Programmers:
  - added five new library functions for handling filetype and subtype operations (e.g. getting a file type name from a sub type value or getting a sub type value from a sub type name, etc.)
  - again revised/reworked/updated programmers docs a little
  - added SVOobject flags indicating DISK read and DISK write support (bit 11 and 12)

#### V14.3 (10.8.1996) :

-----

- SVOobjects: - BMP was broken (sorry)
- LIBRARY:
  - startup-code now is 100% written in C
  - removed any remaining Assembler fragments (there's now only an ULONG remaining, which contains 4 Byte "pure" 68k-Code for RawDoFmt)

#### V14.2 (9.8.1996) :

-----

- NEW SVOobjects:
  - Scanner: forgot to explicitly mention for 14.1, that Scanner support had been added
- Docs: added some notes about 020-40 versions and FPU
- Installer-Script: fixed bug

#### V14.1 (8.8.1996) :

-----

NOTE: For this version of the library some changes did take place, which should help to ensure that it will as well run without problems on OS 4.0 and beyond.

Some changes only have been \_completed\_, since these in parts already took place for 13.x, but others are completely new (semaphore system).

Please note, that all of the included sv-modules now will need at least superview.library V14+ to run - which means they won't work with older versions of the library any longer, while older modules nevertheless still will work with superview.library V14+ (although not recommended).

All included third party modules (OPAL.svdriver, MERLIN.svdriver and SQOperators) yet have to be considered being members of the category "older modules".

- SVOOperators:
    - ALL (except third party):
      - now requiring semaphore mechanism of superview.library V14+
      - new startup-code (see there) (PaletteDither, OptimizePalette, RotateFree already did before)
      - using utility.library and new GST (PaletteDither, OptimizePalette, RotateFree already did before)
      - no longer using MEMF\_PUBLIC for large buffers, which should enable usage of VMM without any further settings to be done (`_small_` buffers should not be VMM'ed, anyway)
      - reworked or added progress indicator support where necessary (sometimes may increase speed). Not explicitly mentioned in SVOoperator histories.
    - Dither24Bit: fixed bug (see there)
    - various: see there
  - SVOObjects:
    - ALL (except third party):
      - now requiring semaphore mechanism of superview.library V14+
      - no longer using MEMF\_PUBLIC for large buffers (see SVOOperators)
      - made sure, that `svgfx_Version` field always will be correctly initialized
      - reworked or added progress indicator support where necessary (sometimes may increase speed). Not explicitly mentioned in SVOobject histories.
      - now correctly examining `svo_check` structure, so that the (no longer supplied) dummy file with Clipboard or Device access will no longer be examined by the other svobjects (may e.g. speed up Clipboard access somewhat)
    - TIFF:
      - completely redone using TIFF Library (see there)
      - "slightly" grew in size
    - JPEG:
      - fixed Enforcer hit (and more, see there)
    - various: see there
  - SVDDrivers:
    - ALL (except third party):
      - now requiring semaphore mechanism of superview.library V14+
    - EGS7, PicassoII, Retina:
      - new startup-code (see there)
    - AGA, CyberGraphics:
      - (see there)
-

- NEW SVOjects: - Limbo: forgot to explicitly mention for 13.3, that Limbo 4.0 (-> Limbo) support had been added
- Support-Lib: - new version; added new function
- LIBRARY:
  - forgot to bump version from 13.2 to 13.3. Now we're at 14.1... ;-)
  - now requires superviewsupport.library V7+
  - internal Datatype module:
    - now uses SVSUP\_AllocGfxBuffer()
    - now supports svgfx\_NativeDIPF (v2) (still replaces old viewmode)
    - now tries to detect and mark HAM, EHB, etc. (oops)
  - (don't blame me, if it doesn't work - at least it's somewhat better than before ;-)
  - there was memory loss at library-expunge time, since some internal structures had not been freed (perhaps because I did not properly set up the new startup-code)
  - slightly changed module expunge mechanism
  - now offering signal semaphores for determination, whether library is still in initialization state or attempts to reload modules (thus SVOjects, SVDdrivers and SVOperators no longer do need to access ExecBase lists) (those modules will then request V14+)
  - bumped version to V14
  - now requires at least (see Programmes note below):
    - SVOjects V4+
    - SVDdrivers V2+
    - SVOperators V3+
  - added rudimentary support for devices like scanners, etc. (see "Programmers")
- Documentation:
  - fixed SVPrefs (Preferences) documentation
  - moved more external files to this Guide (saves some space because of less icons needed, compresses better with LHA - AND is easier to be read...)
  - perhaps no one needs to know about the "SampleConfigs" directory, so removed the icon
- Programmers: - V14 NOW IS THE LOWEST SUPPORTED LIBRARY VERSION (LIBRARY REQUEST MINIMUM IS V14)

While checking the includes, autodocs, etc. again, I found, that some stuff was not very well (or wrongly) documented, although my example sources did show it right.

Now docs and includes have safely been rewritten and all (maybe obsolete or wrong) references



to older library versions have been removed.

Special note for developers of external modules: although not documented this way (the sample sources did it right), all linking nodes of external modules have to be AllocVec'ed and of type MEMF\_PUBLIC (since these maybe do live longer than the modules themselves and may become "recycled").

As already mentioned before: as long as complying to the example sources, you should always have done it the right way perhaps.

FOR THE ABOVE REASONS AND TO ABANDON SOME OBSOLETE MODULES, MODULE REQUIREMENTS HAVE BEEN BUMPED TO AT LEAST:

- SVOjects V3+
- SVDivers V2+
- SVOperators V4+

All included third party modules will continue working with that restriction, but any obsolete, redundant or unauthorized modules based on the last official V11.7 developer docs will perhaps no longer work.

- added new SVOject medium SV\_MEDIUM\_DEVICE (additionally to SV\_MEDIUM\_DISK and \_CLIP), which will allow selective "loading" from and "saving" to specific devices, like e.g. scanners (former "future" parameter of SVL\_Init...Clip).

- FINAL NOTE: Uhm...well.. simply lost track ;-)

V13.3 (27.7.1996) :

- NEW SVOjects:
  - PBM: reads/writes IFF-PBM files as written by the PC-version of Deluxe Paint II (Clipboard supported)
  - GPlot: reads CGM metafile vector graphics by external CGM->Postscript(TM) conversion via GPlot. Similar to AmiFIG/fig support
- SVOjects:
  - CyberDataType:
    - now also works with cyberncg.library and without CyberGraphX
    - could have crashed at startup-time, when picture.datatype V43 not already had been loaded (Ramlib-process problem)
  - PNM: - enhanced in some way (see there)
  - PNG: - updated zlib and pnglib
    - fixed small grayscale bug
    - and more (see there)
  - GhostScript, AmiFIG (new GPlot as well):
    - added possibility to

- "disable-and-replace-by-other-svobject"
  - by STATUS controlpad entry
  - fixed small bug (see there)
- GhostScript: added note about Ghostscript 3.53  
GS\_FONTPATH bug (to be fixed in 4.0)
- SVPrefs:
  - updated (see there)
  - revised GUI
  - (-> Simon Edwards)
- PackerSupport:
  - reorganized directory
  - updated Unpack.svobject (see there)
- LIBRARY:
  - startup-code: fixed de-initialization routine  
(order of library closing)
- Locale:
  - applied some fixes to the swedish localization  
(-> Patrik Nydensten)
- Documentation:
  - the WWW site is no longer available at the moment
  - updated some svobject docs
  - forgot to explicitly mention with V13.2,  
that ILBM now optionally supports extraction  
of ANIM's first ILBM body chunk
  - search&replace-changed terminology used  
in some files (spelling, etc.)
  - merged SVPrefs.doc with guidefile
  - removed obsolete stuff from documentation
  - completely rewrote Notes And Hints Section,  
as well as some other older parts of this  
guide file. Now again worth to be read.
  - updated Simon's http-page URL
  - SuperView HTTP and FTP offline :-(
  - clarified some of the explanations in  
the "Requirements" section
  - !! added english translation of the  
!! Terry Pratchett quotation in the  
!! header page of this guide-file... ;-))

#### V13.2 (18.6.1996) :

- LIBRARY:
  - rewrote startup-code: it was already really  
safe before, but now it has been switched  
from 100% asm to mixed Asm/C (that rocksolid one).  
Also changed order of initialization modules.  
Smarter now. Saved some 100 bytes ;-)
- NEW SVObjects:
  - CorelDraw (TM) CDR Previews:
    - CDR.svobject reads preview pictures of  
of CorelDraw .CDR files
  - Ghostscript:
    - GhostScript.svobject allows importing  
Postscript (TM) vector graphics as bitmap  
using an available Ghostscript installation.  
PDF is also basically supported.
  - AmiFIG (fig):
    - AmiFIG.svobject uses an installed "fig2dev"

(from the AmiFIG port) to create Postscript (TM) files from FIG vector drawings - which then can be read by GhostScript.svobject.  
Thus you'll need both to be installed and configured.

- SVOobjects:
  - BMP:
    - BMP.svobject now also reads RIFF-BMP (DIB)
  - EPS:
    - now optionally allows extraction of Postscript (TM) part of EPS files, which requires Ghostscript and GhostScript.svobject being correctly installed and configured
    - also writes Postscript (TM) now in form of EPS files without trailers (no TIFF) from 24 bit sources
  - GIF, PICT:
    - fixed mem trash bug (4/8 bytes)
  - GIF:
    - now allows writing of transparent GIFs (possible with GIF89a only); useful for creation of HTML pages (-> Patrik Nydensten, Sven Drieling)
  - ILBM:
    - (see there)
  - ACBM, ILBM, DEEP, YUVN:
    - fixed chunk recognition (may have read beyond ID array size)
    - changed/fixed Clipboard file recognition
  - PICT, QRT:
    - (see there)
  - PNG:
    - (see there)
  - DEEP, SGI, SunRaster, Targa, WPG, MAC, FBM, IMG, C64, PNM, Pictor, WinIcon:
    - added new, rock-solid startup code
  - DEEP, SGI, SunRaster, ILBM, C-Source, ACBM, Targa, EPS, YUVN, QRT, FastILBM24, PCD, (UtahRLE):
    - fixed bug in initialization code (may have caused crashes when flushing/reloading)
- SVOoperators:
  - CallPNM: (see there)
- Localization:
  - updated italian "superviewlib.catalog"
  - Thanks to Alessandro Basso for that one !!
- OnlineSupport:
  - SKYNET offline as with 31 May 1996!
  - GM-Box offline as with 31 June 1996!
  - removed reference to n2usx.sauerland.de domain
  - NEW Support BBS and Z-Netz-EMail: CHAOS-Box (domain COB.wwbnet.de)
  - new Fido address (mail to old address will be redirected to the new one)
- Documentation:
  - updated information where necessary

- added some important information to "Requirements" section, so that the needed configuration efforts for Ghostscript, AmiFIG, picture.datatype V43 and so on becomes more obviously.

V13.1 (16.4.1996) :

- 
- LIBRARY:
    - added the concept of "exclusion" of single modules from the startup initialization of the library by user's choice. SVOobjects, SVDdrivers and/or SVOoperators not to be loaded each time can be "excluded" by adding their names to specific configuration files. May increase speed on startup, save memory and reduce the need to call "avail flush" or actually delete modules from their Libs subdirectories (see documentation of "SVPrefs").
    - Has some disadvantages (excluded modules won't be recognized until again - maybe temporarily - loaded by hand or re-included), but these don't make anything worse, since this feature can most simply be ignored.
  - Preferences:
    - added GUI support for "exclusion" model of modules
  - NEW SVOobjects:
    - C-Source.svobject creates C-Source code in chunky/RGB notation from any graphics (<= 8 Bit and HAM as chunky, 24 bit as RGB)
  - SVOobjects:
    - GIF, TIFF:
      - added LZW license check to GIF and LZW-TIFF
    - ILBM:
      - now finally, ultimatively, definitely, absolutely fixed ILBM buffersize calculation stuff (simply had been blind - thanks to -> Simon Edwards)
      - fixed mungwall hits when writing 24 bit IFF-ILBMs (-> Simon Edwards)
  - SVSupport-Lib:
    - fixed grabbing of interlaved screens with ScreenToBitPlane() (-> Simon Edwards)
  - Programmers:
    - fixed superview.fd: GetTypeExtension (-> Simon Edwards)
    - added new function SVL\_SetDTMode(), which allows enabling/disabling datatypes support (if enabled, it also supersedes the LIBRARY controlpad entry "ANYDATATYPES" by specifying own settings)
  - Documentation:
    - updated Simon's EMail address
    - OnlineSupport:
      - SNOWBALL is offline!
      - gm-box/mediaserve domain canceled
      - skynet domain changed
      - new WWW page and FTP address

- with SuperView-Support (GM-Box)
- added other doc files as external links
- rewrote SQ-Operators.guide
- added more links to this document and updated information where necessary

```
*****
* Revision history entries below V13.1 have been deleted          *
*           to save diskpace !                                     *
*****
```

## 1.27 Printer.svobject

© 1996 by Andreas R. Kleinert.  
 FREeware. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.1  
 Release Date : 15.08.1996

Description  
 ~~~~~

Printer.svobject is an external library module for the superview.library.

It supports printing from any Screen (upto 256 Colors)  
 or graphics buffer (upto 256 Colors or 24 Bit).

Currently, all 24 Bit images will be dithered to 256 colors  
 before printing (using the "Dither24Bit" operator).

Usually you get best results, when first applying "AnyTo24Bit"  
 to a 256 color graphics and then call Printer.svobject  
 from your application's DEVICE SVObject list.

ControlPad-Switches  
 ~~~~~

```
ControlPad-Name      : "ENV:SuperView-Library/Printer.controlpad"
ControlPad-Commands : - DENSITY=<density value for printing quality>
                      ; Amiga's printer device does allow seven
                      ; modes of printing density (1-7)
                      ; default is: 3
```

History  
 ~~~~~

V4.1 (15.08.1996) :

- first version

## 1.28 Scanner.svobject

© 1996 by Andreas R. Kleinert.  
 FREeware. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.3  
Release Date : 14.09.1996

#### Description

~~~~~

Scanner.svobject is an external library module for the superview.library.

It supports scanning images by using Andreas Günther's scanner.device,  
as supplied with the ScanQuix software package.

Currently, all scanned images are imported as 24 Bit.

#### Credits

~~~~~

ScanQuix currently supports:

- Epson (GT-4000, GT-5000, GT-6000, GT-6500,  
GT-8000, GT-8500, GT-9000)  
SCSI, serial or parallel
- HP (ScanJet 2c, 2cx, 3c, 4c, 4p)  
SCSI
- Mustek (Paragon 600, 600 SP, 800 SP,  
1200 SP)  
SCSI

It is available from:

RBM Computertechnik  
Bernd Rudolf  
Kleinenberger Weg 2a  
D-33100 Paderborn Phone: +49-5251-640646  
Germany Fax: +49-5251-640655

For more information, please contact this company.

#### ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/Scanner.controlpad"  
ControlPad-Commands : - PUBSCREEN\_NAME=<PubScreenName>  
; where the scanner drivers should open  
; their windows (if not set or not available,  
; the default Public Screen will be used).

#### History

~~~~~

V4.3 (14.09.1996) :

- fixed error string on open device failure
- major bug: did not set svgfx\_BufferSize
- added controlpad for specifying a PubScreen name

V4.2 (18.08.1996) :

- skipped

V4.1 (08.08.1996) :

- first version
-

## 1.29 AmiFIG.svobject

© 1996 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

### Description

~~~~~

AmiFIG.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support, a working GhostScript.svobject and an installed fig2dev executeable.

It supports importing AmiFIG/Xfig vector drawings as bitmapped graphics. This is done by using fig2dev for conversion to Postscript (TM) files, which then will be parsed using a working Ghostscript installation with GhostScript.svobject.

AmiFIG.svobject will create a temporary file, which then will be parsed through superview.library again.

### Reading :

(Any FIG drawings as long as the resulting Postscript (TM) commands are readable for Ghostscript).

### Configuration

~~~~~

This FIG support module has been tested with:

- the AmiFIG 1.1 port from AmiNet: AmiFIG is (C) 1996 Andreas Schmidt AND NOT YET, because of missing fig2dev:
- the Xfig 1.9d amiga port from AmiNet, which has been done by Terje Pedersen (terjepe@login.eunet.no)

At first, you have to correctly install and configure Ghostscript and GhostScript.svobject.

After that you have to install AmiFIG/Xfig and either supply an AmigaDOS search path to a directory, where the executable "fig2dev" is stored (default), or explicetely specify path plus name of the program by the controlpad entry AMIFIG\_PATH (see below).

### Where to DOWNLOAD from

~~~~~

AmiNet:gfx/edit/Amifig#?.LHA (#?=version and CPU)

### ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/AmiFIG.controlpad"

ControlPad-Commands : - AMIFIG\_PATH=<fig2dev command path plus name>  
; how fig2dev is to be called

```

; e.g. AMIFIG_PATH=Work:AmiFIG/fig2dev
; default is: fig2dev
- STATUS=<ENABLED|DISABLED>
; allows to disable this module - for example
; to be able to use an other, program-specific
; import-module for the same file format

```

#### History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- now correctly handles SVOCheckFile

V3.2 (23.07.1996) :

- added possibility to disable this module, if there's an other, program-specific sobject you would like to use instead (xFIG ?)
- changed file format extension from ".ps" to ".fig", although perhaps never of practical use (since not write-able)
- fixed error return mechanism for SVO\_Show (accidentally correct ?)

V3.1 (07.06.1996) :

- first version

## 1.30 Degas.svobject

© 1996 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.1

Release Date : 21.08.1996

#### Description

~~~~~

Degas.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support.

It supports reading of Atari Degas files.

In detail these are :

Reading :

- uncompressed Degas PI1 320x200, 16 colors
- uncompressed Degas PI2 640x200, 4 colors
- uncompressed Degas PI3 640x400, 2 colors

#### History

~~~~~



V4.1 (21.08.1996) :

- first version

## 1.31 ILBM.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.3

Release Date : 16.09.1996

Description

~~~~~

ILBM.svobject is an external library module for the superview.library, which needs any SVDriver with Bitplane-Support.

It supports reading and writing of IFF-ILBM files.

In detail these are :

Reading :

- IFF-ILBM uncompressed 1..8/24 bit
- IFF-ILBM CmpByteRun1 compressed 1..8/24 bit

Writing :

- IFF-ILBM uncompressed 1..8/24 bit
- IFF-ILBM CmpByteRun1 compressed 1..8/24 bit

Reading/writing from/to ClipBoard is supported.

ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/ILBM.controlpad"

ControlPad-Commands : - ANIM\_BODIES

```

; if set, ANIM files' ILBM-BODY chunks
; (first frame) will be extracted when
; such a file is encountered. Otherwise
; ANIM.datatype may do that task later.
```

History

~~~~~

V4.3 (16.09.1996) :

- version had not been bumped to V4.2
- now rejects palette files (no body, no gfx) already when file-checking (-> Jürgen Schäfer)
- generates grayscale palette for files without color table

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.6 (08.06.1996) :

- fixed bug in initialization code
- fixed chunk recognition (may have read beyond ID array size)
- fixed Clipboard file recognition (BMHD had to be first chunk)
- now supports reading of ANIM files' first frame (optional)

V3.5 (13.04.1996) :

- now finally, ultimately, absolutely fixed ILBM buffersize calculation stuff (simply had been blind - thanks to -> Simon Edwards)
- when saving 24 bit ILBMs, the buffer for 24 bit oneplane to bitplane conversion was not correctly aligned, thus a little bit to small (mung wall hits) - the conversion buffer for reading on the other hand was a little bit to large, which did not actually matter. Fixed. (-> Simon Edwards)

V3.4 (14.03.1996) :

- IMPORTANT fix: argh! ILBM just had the same, nasty "not more than 256 bytes per line" bug as PCX had in former versions. It was so simple, that I didn't even saw it. Fixed now. Side-effect: buffer perhaps now always will be in available 32 Bit Fast Ram :-)
- Concerned reading \_and\_ writing.
- no longer searches for 24 bit viewmodes (ask for 8 Bit)
- completely rewrote decoding routines for unpacked (interleaved -> planebuffer) and packed (CmpByteRun1 -> planebuffer) routines. No longer using mixed UWORDS and ULONGs, only ULONGs and two UBYTES for counting, loops, etc. This may slow down processing on plain 68000 systems, but will a) increase speed on 680x0 and b) reduce possible casting errors and/or overflow errors caused by the compiler or utility library. Hope so.
- fixed progress indicator update intervals (reading and writing)
- writing: no longer depends on valid source (or temporary) svgfx\_BufferSize (would have caused mungwall hits when wrongly initialized). Also fixed buffersize calculation.

```
*****
* Revison history entries before V3.4 have been deleted for space reasons *
*****
```

## 1.32 PBM.svobject

© 1996 by Andreas R. Kleinert.  
FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

#### Description

~~~~~

PBM.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support.

It supports reading and writing of IFF-PBM files, which originally only were written by Deluxe Paint II for the PC, when saving chunky 256 Color graphics.

In detail these are :

#### Reading :

- IFF-PBM uncompressed 8 Bit
- IFF-PBM CmpByteRunl compressed 8 Bit

#### Writing :

- IFF-PBM uncompressed 8 Bit (less than 256 colors will be
- IFF-PBM CmpByteRunl compressed 8 Bit increased to 256 colors)

Reading/writing from/to ClipBoard is supported.

#### History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.1 (05.07.1996) :

- first version

## 1.33 ACBM.svobject

© 1994-96 by Andreas R. Kleinert.  
FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.3  
Release Date : 16.09.1996

---

## Description

~~~~~

ACBM.svobject is an external library module for the superview.library, which needs any SVDriver with Bitplane-Support.

It supports reading and writing of IFF-ACBM files.

In detail these are :

## Reading :

- IFF-ACBM uncompressed 1..8 Bit

## Writing :

- IFF-ACBM uncompressed 1..8 Bit

Reading/writing from/to ClipBoard is supported.

## History

~~~~~

## V4.3 (16.09.1996) :

- now rejects palette files (no body, no gfx) already when file-checking (-> Jürgen Schäfer)
- generates grayscale palette for files without color table
- ClipBoard file recognition checked for BODY instead of ABIT. Fixed.

## V4.2 (29.08.1996) :

- now has new disk read/write flags set
- fixed word-alignment when reading and writing (this bug was even older than superview.library ;-)  
side-effect: uses quite as much memory as IFF-ILBM, now
- would not have returned an error, when detecting unknown compression types
- would not have returned an error in a specific low-memory situation

## V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

## V3.4 (08.06.1996) :

- fixed bug in initialization code
- fixed chunk recognition (may have read beyond ID array size)
- fixed Clipboard file recognition (BMHD had to be first chunk)

## V3.3 (27.01.1996) :

- added REV. 2 of new, rock-solid startup code
- now will reject graphics with depths greater than 8 (e.g. 24 bit graphics) - which had not been supported anyway, but might have caused crashes - when reading and writing

- using new GST

V3.2 (24.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (07.10.1995) :

- upgraded to V3 specs

```
*****
* Revision history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.34 Datatypes support

Loading Graphics via DataTypes under OS3

=====

In General

-----

The library generally also accesses OS3-Datatypes, if none of the installed SVOobjects was able to identify a specific graphics file format.

Usually by default only Datatypes of class "picture" will be accepted.

This means, no Samples, Texts, AmigaGuide File or whatever else will be loaded and displayed via superview.library - by default.

But it is possible to enable these also - see configuration section for more information on how to change settings - so that for example introduction pictures of ANIMs may as well be loaded.

Specific Problems

-----

As you may have expected, it is as well possible to replace single SVOobjects by the corresponding Datatypes in case you think that the Datatype does work better or more reliable.

For temporarily enabling this, you would just have to "Remove" the SVOobject and load the Datatype.

For permanently enabling this, you would have to delete the SVOobject from "LIBS:svobjects".

But remember, that DTs in general cannot export 24 bit Data and are not able to save any graphics (other than as IFF-ILBM).

Ralph Schmidt's picture.datatype V43 does allow loading of 24 bit pictures by enhancing the datatype system with additional functionality. To be able using these features with SuperView, you need a registered CyberGraphX installation and CyberDataType.

On the other hand DO NOT TRY to run picture.datatype V43 without CyberDataType being installed. SuperViewLibrary's internal Datatypes

reader likes accessing bitmaps directly - but CyberGraphX bitmaps may be somewhat stranger than the usual ones. Precisely, SuperViewLibrary does touch DT bitmaps only, when trying to extract these as GfxBuffers.

## 1.35 PCX.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

### Description

~~~~~

PCX.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support.

It supports reading and writing of PCX files.  
In detail these are :

### Reading :

2-256 Colors, 24 bit. In detail:

- PCX V2.5 upto 16 Colors (supports 4/8 color EGA/VGA palette)
- PCX V2.8a upto 16 Colors (supports 4/8 color EGA/VGA palette)
- PCX V2.8b upto 16 GreyScales
- PCX V3.00 upto 256 Colors
- PCX V2.5 - V3.00 with 24 bit Data (see Remarks !)

### Writing :

Depending on the Colordepth of the source the following is written :

Source Colors	Version	Type	Destination Colors
2 .. 16	PCX V2.8a	planar bitmap	16
32 .. 256	PCX V3.00	chunky pixel	256
(24 bit)	PCX V3.00	RGB planar	(24 bit)

For 2-256 Colors It is always tried, to write the files RLE-encoded, but if encoding is ineffective (output data nearly as large or even larger than input data), the files will be written unencoded. 24 bit files will currently only be written unencoded.

### History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+

- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.4 (14.3.1996) :

- fixed safety buffer calculation (as for ILBM, just with 64)

V3.3 (20.12.1995) :

- could not handle planar graphics (b&w or upto 16 Colors) with a width greater than 2048 (256\*8 -> buffer overflow). Fixed.  
(-> Jürgen Schäfer)
- added REV. 2 of new, rock-solid startup code
- no longer causes crashes
- using new GST

V3.2 (24.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (07.10.1995) :

- upgraded to V3 specs

```
*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.36 SVG.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 15.09.1996

Description

~~~~~

SVG.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support.

It supports reading and writing of SVG Graphics files, which is SuperView-Library's own graphics file format with XPK support.

In detail these are :

Reading :

- SVG ONEPLANE 8/24 bit  
BITPLANE 1..8 Bit

Writing :

```
- SVG ONEPLANE 8/24 bit
  BITPLANE 1..8 Bit
```

#### ControlPad-Switches

```
~~~~~
```

```
ControlPad-Name      : "ENV:SuperView-Library/SVG.controlpad"
```

```
ControlPad-Commands : - PACKMETHOD=<xxxx>
                        ; if this one is specified, it is tried to
                        ; pack the resulting file with the specified
                        ; XPK-Packer. If this fails, the file keeps
                        ; unpacked.
```

#### File Format Information

```
~~~~~
```

On "AmiNet:util/dtype" there's a separate package available, which consists of a "SVG.datatype" for applications not using SuperView-Library, the description of the FileFormat and a conversion tool "AnyToSVG" (using Datatypes).

Some excerpts from there:

Pictures in the "SVG Graphics File Format" consist of two parts: a header and an attached xpk-packed or unpacked data file. Both parts are put into one single file.

#### Construction:

0x00	ID	UBYTE[18]	"SVG Graphics File"
0x12	Version	UWORD	always 1 yet
0x14	GfxDataOffset	ULONG	header length (depends on version)
0x18	LeftEdge	ULONG	
0x1c	TopEdge	ULONG	
0x20	Width	ULONG	
0x24	Height	ULONG	
0x28	ColorDepth	ULONG	actually available colors
0x3c	ViewMode32	ULONG	32 Bit ViewMode
0x40	PixelBits	UBYTE	1, 8, 24 (future: 16, 32)
0x41	PixelPlanes	UBYTE	# of planes with PixelBits
0x42	BytesPerLine	ULONG	bpl of a PixelPlane
0x46	ColorMap	UBYTE [256][3]	empty, if > 256 Colors

After that either follows XPK compressed data or uncompressed data, which can be detected by the leading chars "XPK" or "PP20" for packed data at GfxDataOffset (relative to beginning of the file).

#### History

```
~~~~~
```

V4.2 (15.09.1996) :

- now has new disk read/write flags set
- was missing in V15.1 (declared to have been written on 18.08.96)

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+



- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.4 (21.12.1995) :

- added REV. 2 of new, rock-solid startup code

V3.3 (10.12.1995) :

- added new, rock-solid startup code
- no longer causes crashes
- changed some includes and so on
- now opens "xpkmaster.library" just when needed and not when initializing module. This should have two effects:
  - if you want to save packed data, but there's no library, it will be saved unpacked instead
  - initialization time should be much shorter, so that possibly the crashes occuring on 060 systems should no longer occur (hopefully)
- the xpk-libraries may be flushed from memory completely, when no longer actually being needed and used
- opening the svobject does no longer depend on the presence of xpk- and or powerpacker-library (so unpacked files may as well be read/written on systems without these)

V3.2 (24.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (07.10.1995) :

- upgraded to V3 specs

V2.1 (24.09.1995) :

- replaced SVO.svobject by this one, which is smarter and works a little bit more transparent for the user. Needs VMEM: assignment.

## 1.37 GIF.svobject

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits" and LZW notes under "Copyright".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.3  
Release Date : 07.09.1996

Description  
~~~~~

GIF.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support.

For using this module, LZW support must be globally enabled.

---

It supports reading and writing of GIF files (87a and 89a derivatives are supported, where additional features of 89a will just be ignored).

In detail these are :

Reading :

- GIF 87a
- GIF 89a

Writing :

- GIF 87a
- GIF 89a

| Source Colors | Version    | Type         | Destination Colors |
|---------------|------------|--------------|--------------------|
| 2 .. 256      | GIF 87/89a | chunky pixel | 2 .. 256           |

Bugs and other Problems

~~~~~

Decoding is reentrant, encoding isn't:

Please note, that the Encoding Routines of the Library are not (yet) fully re-entrant, so that only one Task may Encode a picture at a time. The Library itself manages it, that the other Task has to Delay() until the Encoding Routines are "free" again.

This makes NOT REALLY A MATTER, because Encoding takes so much of the processor's time, that it is NOT EFFICIENT to ENCODE TWO OR MORE pictures AT THE SAME TIME ANYWAY !

ControlPad-Switches

~~~~~

```
ControlPad-Name      : "ENV:SuperView-Library/GIF.controlpad"
ControlPad-Commands : - GIF89A_TRANSPARENT_COLOR = <0..255>
                      ; Default color to be enabled and set
                      ; (graphics control extension block)
                      ; Default: (not active)
```

History :

~~~~~

V4.3 (07.09.1996) :

- writing transparent GIF89a files did not work due to a \*stupid\* bug

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.5 (15.6.1996) :

- fixed bug, which may have caused upto 4 bytes of the library (base) to possibly be overwritten
- now allows writing of transparent GIFs (possible with GIF89a only); useful for creation of HTML pages. Transparent color is both, enabled and set, by a new controlpad entry.  
(-> Patrik Nydensten, Sven Drieling)

V3.4 (16.3.1996) :

- added LZW license check

```
*****
* Revision history entries before V3.4 have been deleted for space reasons *
*****
```

## 1.38 GPlot.svobject

© 1996 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

### Description

~~~~~

GPlot.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support, a working GhostScript.svobject and an installed GPlot executeable.

It supports importing CGM metafile vector drawings as bitmapped graphics. This is done by using gplot for conversion to Postscript (TM) files, which then will be parsed using a working Ghostscript installation with GhostScript.svobject.

GPlot.svobject will create a temporary file, which then will be parsed through superview.library again.

### Reading :

(Any CGM drawings as long as the resulting Postscript (TM) commands are readable for Ghostscript - especially the Times-Roman font has to be available for Ghostscript).

### Configuration

~~~~~

This CGM/GPlot support module has been tested with the GPlot 4.3b2a1 port from AmiNet:gfx/conv. GPlot is (C) 1987 by the Pittsburgh Su Center, the original version had been done by Phil Andres, Amiga port by Michael Cheng in 1996.

At first, you have to correctly install and configure Ghostscript and GhostScript.svobject.

After that you have to install GPlot and either supply an AmigaDOS search path to a directory, where the executable "GPlot" is stored (default), or explicitly specify path plus name of the program by the controlpad entry GPLOT\_PATH (see below).

Please note, that the resulting Postscript (TM) files, which then will be passed through Ghostscript, definitely do contain a reference to the font "Times-Roman" (in Ghostscript's FONTMAP this is redirected to the font file ptmr.gsf, which has to be present in the font directory(ies) specified in the environment variable GS\_FONTPATH). With the Ghostscript version I had for testing, the font support did not work, due to a bug (or configuration fault ;- ) of Ghostscript.

So, please make sure, that this font is present - otherwise an error message like "file not found" may appear.

Where to DOWNLOAD from  
 ~~~~~  
 AmiNet:gfx/conv/gplot.LHA

ControlPad-Switches  
 ~~~~~  
 ControlPad-Name : "ENV:SuperView-Library/GPlot.controlpad"  
 ControlPad-Commands : - GPLOT\_PATH=<gplot command path plus name>  
                           ; how gplot is to be called  
                           ; e.g. GPLOT\_PATH=Work:GPlot/GPlot  
                           ; default is: GPlot  
       - STATUS=<ENABLED|DISABLED>  
           ; allows to disable this module - for example  
           ; to be able to use an other, program-specific  
           ; import-module for the same file format

History  
 ~~~~~  
 V4.2 (18.08.1996) :  
     - now has new disk read/write flags set  
 V4.1 (06.08.1996) :  
     - now uses/requires semaphore system of superview.library V14+  
     - now correctly handles SVOCheckFile  
 V3.1 (16.07.1996) :  
     - first version

## 1.39 BMP.svobject

© 1994-96 by Andreas R. Kleinert.  
 FREeware. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.3

Release Date : 28.08.1996

#### Description

~~~~~

BMP.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support.

It supports reading and writing of Windows (TM) and OS/2 (TM) BitMap files (BMP). Not all derivatives might be supported, though. Extraction of Bitmaps from RIFF-BMP files (DIB) also is possible.

In detail these are :

#### Reading :

- unencoded BMP "wallpapers" in 1, 4, 8, 24 bit ColorDepth.
- RLE-encoded BMP "wallpapers" in 4, 8 Bit ColorDepth.

#### Writing :

Depending on the Colordepth of the source the following is written :

Source Colors	Version	Type	Destination Colors
2	BMP (misc)	packed chk. pix.	2
4 .. 16	BMP (misc)	packed chk. pix.	16
32 .. 256	BMP (misc)	chunky pixel	256
(24 bit)	BMP (misc)	RGB pixel	(24 bit)

#### History

~~~~~

V4.3 (28.08.1996) :

- now has new disk read/write flags set
- there was a bug in writing the BMP header, which caused any programs correctly examining the header (including SuperView) to distort the written bitmap because of a wrong offset

V4.2 (10.08.1996) :

- V4.1 was broken

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.4 (29.04.1996) :

- now can read RIFF-BMP files (DIB)

\*\*\*\*\*  
 \* Revision history entries before V3.4 have been deleted for space reasons \*

\*\*\*\*\*

## 1.40 WinIcon.svobject

© 1994-96 by Andreas R. Kleinert.

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

Description

~~~~~

WinIcon.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support.

It supports reading of W\*nd\*ws V3.x (TM) Icon files.

In detail these are :

Reading :

- WinIcon files with exactly 16 Colors  
(more than one Icon per file should work, but hasn't been tested yet.)

History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.3 (09.06.1996) :

- added new, rock-solid startup code

\*\*\*\*\*  
 \* Revision history entries before V3.3 have been deleted for space reasons \*  
 \*\*\*\*\*

## 1.41 FBM.svobject

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

## Description

~~~~~

FBM.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support.

It supports reading and writing of FBM (\*nix) files.

In detail these are :

## Reading :

- FBM 8 Bit x 1 Plane: 2..256 Colors/256 Greyscales
- FBM 8 Bit x 3 Planes: 24 bit

## Writing :

- FBM 8 Bit x 1 Plane: 2..256 Colors/256 Greyscales
- FBM 8 Bit x 3 Planes: 24 bit

Following to the specifications, an 2..128 Colors file takes as much space as an 256 Colors file : always 8 Bits are written.

Only the ColorMap is sized differently for different ColorDepths.

## History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.3 (09.06.1996) :

- added new, rock-solid startup code

```
*****
* Revison history entries before V3.3 have been deleted for space reasons *
*****
```

## 1.42 Limbo.svobject

© 1996 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.3

Release Date : 16.09.1996

\*\*\* THIS ONE IS REALLY EXPERIMENTAL - IT WORKS, BUT REQUIRES

A LOT OF MEMORY AND A FAST PROCESSOR.  
DO NOT SAVE IMPORTANT DATA AS LIMBO/LMB. IT'S HIGHLY LOSSY! \*\*\*

#### Description

~~~~~

Limbo.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support and an installed Limbo 4.0 executeable.

It supports importing Limbo 4.0 fractal-compressed bitmapped graphics. This is done by using Limbo for conversion of LMB files to temporary PNM files, which then will be parsed through superview.library again. Exporting is supported by creating temporary PNM files, which then will be converted to LMB files via Limbo.

A VMEM: assignment is needed to be present to allow doing the temporary file and directory management.

So it supports reading and writing of Limbo 4.0 files.  
In detail these are :

#### Reading :

(Limbo 4.0 fractal-compressed files, as far as supported by the used Limbo program.)

#### Writing :

(Any input will be converted to 24 Bit, which then will be converted to Limbo 4.0 fractal-compressed files, by the Limbo program.)

#### Configuration

~~~~~

This Limbo support module has been tested with the Limbo 4.0 port from AmiNet:gfx/conv, which was based on a version developed by Carsten Frigaard, Jess Gade, Thomas Therp Hemmingsen and Torben Sand in 1993/94 on Aalborg University, Denmark.

At first you have to install Limbo by simply copying it into a directory where you like it to reside.

You also have to either supply an AmigaDOS search path to a directory, where the executable "Limbo.68000" is stored (default), or explicitly specify path plus name of the program by the controlpad entry LIMBO\_PATH (see below).

#### Where to DOWNLOAD from

~~~~~

AmiNet:gfx/conv/Limbo.LHA

#### ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/Limbo.controlpad"

ControlPad-Commands : - LIMBO\_PATH=<Limbo command path plus name>  
; how Limbo is to be called



```

; e.g. LIMBO_PATH=Work:Limbo/Limbo.68030.881
; default is: Limbo.68000
- STATUS=<ENABLED|DISABLED>
; allows to disable this module - for example
; to be able to use an other, program-specific
; import-module for the same file format
;
; DECODING settings
; *****
- DECODE_ITERATIONS=<number>
; option -i of Limbo 4.0
; default is: 6
- EXPANSION_LEVEL=<number>
; option -l of Limbo 4.0
; default is: 0
- DECODE_VERBOSE=<ON|OFF>
; Should decoding output be sent to stdio ?
; default is: OFF
;
; ENCODING settings
; *****
- EXPANSION_LEVEL=<number>
; option -l of Limbo 4.0
; default is: 0
- MIN_RANGE_BLOCKSIZE=<number>
; option -b of Limbo 4.0
; default is: 4
- DOMAIN_SLIDE_BLOCKSIZE=<number>
; option -d of Limbo 4.0
; default is: 8
- MIN_COMPARE_DOMAIN_BLOCKS=<number>
; option -m of Limbo 4.0
; default is: 4
- MAX_COMPARE_DOMAIN_BLOCKS=<number>
; option -s of Limbo 4.0
; default is: 10
- FEATURE_SPACE_DIMS=<number>
; option -f of Limbo 4.0
; default is: 3
- GRID_DIV_RES=<number>
; option -r of Limbo 4.0
; default is: 1000
- QUADTREE_LEVEL=<number>
; option -n of Limbo 4.0
; default is: 0
- TRESHOLD_SPLIT=<number>
; option -Tm of Limbo 4.0
; default is: 100
- TRESHOLD_CLASS=<number>
; option -Te of Limbo 4.0
; default is: 50
- TRESHOLD_POST=<number>
; option -Tp of Limbo 4.0
; default is: -1
- ENCODE_VERBOSE=<ON|OFF>
; Should encoding output be sent to stdio ?
; (recommended, since encoding is

```

---

```
; _extremely slow_)
; default is: ON
```

#### History

~~~~~

V4.3 (16.09.1996) :

- file information did not work (no requester)
- V4.2 had V4.1 in the libid string

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- now correctly handles SVOCheckFile

V3.1 (26.07.1996) :

- first version. Still as experimental as Limbo itself.

## 1.43 PNM.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

#### Description

~~~~~

PNM.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support.

It supports reading and writing of PNM (\*nix) files.  
In detail these are :

#### Reading :

All binary (non-ASCII) derivatives of the PNM format will be read (P4/P5/P6), non-binary (ASCII) derivatives are not supported (P1/P2/P3).

- PBM (P4) Black & White
- PGM (P5) 256 Grayscales
- PPM (P6) 24 bit TrueColor

#### Writing :

- PGM (P5) 256 Grayscales
- PPM (P6) 24 bit Truecolor

Writing graphics as binary 24 bit PPM data (P6) is the

---

only way to prevent color information from getting lost.

#### ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/PNM.controlpad"

ControlPad-Commands : - SUPPRESS\_HEADER  
                           ; setting this keyword will suppress  
                           ; writing the PNM header for P5/P6  
                           ; files, which actually will generate  
                           ; a RAW data file, which cannot be loaded  
                           ; with superview.library again, but may be  
                           ; used for different purposes  
                           ; Default: (not set)

#### History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.4 (21.07.1996) :

- added write support for PGM (P5) with 256 Grayscales
- added possibility to easily generate RAW data files by controlpad option settings
- could not handle files with comment lines (beginning with "#") in the header - like e.g. generated by Limbo 4.0. Now smartly skipping these lines.
- bad PNM files might have caused an endless loop (never happened, but just detected that)
- now using buffered IO for that byte fiddling (FRead())

\*\*\*\*\*  
 \* Revison history entries before V3.4 have been deleted for space reasons \*  
 \*\*\*\*\*

## 1.44 PNG.svobject

© 1996 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

#### Description

~~~~~

PNG.svobject is an external library module for the superview.library,

which needs any SVDriver with Oneplane-Support.

It supports reading of Portable Network Graphics (PNG - pronounce it "PiNG", if you like ;-) files.

In detail these are :

Reading :

- 8 Bit color mapped files
- True color files (16, 24/32, 48 Bit) as 24 bit  
while: 16 Bit (5:5:5/5:6:5) will be expanded, and  
48 Bit (16:16:16) will be cut down to 24 Bit (8:8:8)

History :

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.3 (16.07.1996) :

- upgraded to zlib V1.0.2 and pnglib V0.88, which should fix some bugs and introduce new and better functionality
- seems to be faster now
- added progress indicator support
- fixed bug in reading grayscale images:  
would not have the correct 256 color grayscale palette set, but instead the (accidental ?) content of the delivered PNG palette structure, which had not been initialized for grayscales

```
*****
* Revison history entries before V3.3 have been deleted for space reasons *
*****
```

## 1.45 C64.svobject

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

Description

~~~~~

C64.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support.

It supports reading of C64 Graphics files (Koala, Doodle).  
In detail these are :

Reading :

Format	Dimensions	Colors
Doodle	320x200	2/16
Koala	160x200 -> 320x200	4/16

History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.3 (09.06.1996) :

- added new, rock-solid startup code

\*\*\*\*\*  
\* Revison history entries before V3.3 have been deleted for space reasons \*  
\*\*\*\*\*

## 1.46 CDR.svobject

© 1996 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

Description

~~~~~

CDR.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support and BMP.svobject to be present.

It supports reading of the previews which may be part of CorelDraw .CDR files (e.g. with version 4) - these are related with plain Windows BMP files, so that CDR.svobject will extract the graphics data, create a temporary file, write it as a BMP file and then parse it through superview.library again.

(See documentation of BMP.svobject for which BMP derivates currently are supported).

Reading :

(Any contained preview, as long as the specific contained BMP derivate is supported by superview.library).

History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.1 (07.06.1996) :

- first version

## 1.47 IMG.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

Description

~~~~~

IMG.svobject is an external library module for the superview.library, which needs any SVDriver with Bitplane-Support.

It supports reading of IMG (GEM/Metafile) files.

In detail these are :

Reading :

- IMG (16 byte header) greyscaled (1..8 bits = 2..256 scales)
- IMG (18 byte header) greyscaled (1..8 bits = 2..256 scales)

History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
  - no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
  - made sure, that always correct svgfx\_Version is set
  - now correctly handles SVOCheckFile
-

V3.3 (09.06.1996) :

- added new, rock-solid startup code

```
*****
* Revision history entries before V3.3 have been deleted for space reasons *
*****
```

## 1.48 TIFF.svobject

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits" and LZW notes under "Copyright".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 24.08.1996

Description

~~~~~

TIFF.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support.

It supports reading and writing of TIFF-files (V5.0).

In detail these are :

Reading :

Compression types (as far as known):

- |                    |   |
|--------------------|---|
| - uncompressed     | - CCITT Group 3 Fax                     |
| - LZW              | - CCITT Group 4 Fax                     |
| - NeXT (2-bit RLE) | - CCITT/3 1D (Huffman RLE)              |
| - Mac PackBits     | - CCITTRLEW (word-aligned uncompressed) |
| - Thunderscan RLE  | - JPEG (decompression not possible)     |

Bit depths:

Pixelbits	Planes	=> resulting Bit Depth
1	1	1
1	x	x (Mac Packbits only)
4	3	12 (imported as 24 Bit)
8	1	8 (256 colors or gray)
8	3	24
8	4	24 (24 Bit + alpha Channel)

Writing :

Depth	written as
<= 256 colors	Motorola (MM), None, 8, 1
(24 bit)	Motorola (MM), None, 8, 3

## History

~~~~~

V4.2 (24.08.1996) :

- the new TIFF reader code did not correctly left-shift the UWORD palette entries of most palette-based pictures (only some uncompressed or Mac Packbit compressed, beeing handled by the workaround-code, would have worked). All palette entries did became black.
- \*Serious\* bug.
- now has new disk read/write flags set

V4.1 (06.08.1996) :

- completely removed my own TIFF reading routines and replaced these by the freely available TIFF Library (only for "unpacked" and MacPackBits there do remain some fragments, since TIFF Library does seem to have some deficites here, so we do use my own routines as a fallback here)
- because of using TIFF Library, TIFF.svobject now became really large (worse), but on the other hand now does (hopefully ;-) support the following compressions when reading: uncompressed, CCITTRLE, CCITTFax3, CCITTFax4, LZW (when enabled), Next (2-bit RLE), CCITTRLEW, Mac PackBits, ThunderScan
- hope, that these changes did not make anything worse, but it simply wasn't possible to make LZW bug-free and add CCITT Fax support without doing so
- also, it now can safely handle 32 Bit graphics (24 Bit plus Alpha Channel) and those 12 Bit pictures (4:4:4 "Truecolor")
- removed 030 version, since 2x200K simply would have been too much
- TIFF.svobject seems to be really fast now
- added progress indicator support
- slightly changed "unpacked" and Mac Packbits (fallback) routines
- added/changed/fixed description of old/new compression types
- writing remains unchanged
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile
- now uses/requires semaphore system of superview.library V14+

V3.6 (16.03.1996) :

- added LZW license check

```
*****
* Revision history entries before V3.6 have been deleted for space reasons *
*****
```

## 1.49 EPS.svobject

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.



Version : 4.2  
Release Date : 18.08.1996

#### Description

~~~~~

EPS.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support and TIFF.svobject (respectively GhostScript.svobject) to be present.

It allows two operations, depending on controlpad preferences settings:

- extraction of the trailers/previews which may be part of EPS(F) files. This usually will be TIFF graphics, but it makes no matter, which file format actually is included, because EPS.svobject will create a temporary file, which then will be parsed through superview.library again.  
(See documentation of TIFF.svobject for which TIFF derivatives currently are supported).
- extraction of the Postscript (TM) part of EPS(F) files, which then, in form of a temporary file, will be parsed through superview.library (and perhaps also GhostScript.svobject) again.

#### Reading :

- Any contained trailers/previews/thumbnails, as long as the specific contained file format - if any - is supported by superview.library
- Postscript (TM) as far as supported by the installed Ghostscript implementation

#### Writing:

- EPS files with Postscript (TM) image content, without TIFF header (only from 24 bit sources -> use the AnyTo24Bit operator).

The graphics will NOT be vectorized - instead a bitmap-like representation of it will be translated into Postscript (TM) format.

#### Remarks

~~~~~

Please note, that the Encoding Routines of the Library are not (yet) fully re-entrant, so that only one Task may Encode a picture at a time. The Library itself manages it, that the other Task has to Delay() until the Encoding Routines are "free" again.

#### ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/EPS.controlpad"  
ControlPad-Commands : - EXTRACTMODE=<HEADER|PS>  
; Determines, whether the preview image or  
; the Postscript (TM) part should be extracted,  
; while parsing the Postscript (TM) part requires  
; a working Ghostscript installation with  
; GhostScript.svobject being correctly configured  
; default is: HEADER

---

- SAVE\_ROTATE=<ON|OFF>
  - ; by default, the picture seems to be rotated
  - ; with Postscript (TM) output of this module.
  - ; Enabling SAVE\_ROTATE will again re-rotate it to
  - ; the original position.
  - ; default is: OFF
- SAVE\_CENTER=<ON|OFF>
  - ; centers the images on the page with the
  - ; desired size of SAVE\_WIDTH x SAVE\_HEIGHT (or not)
  - ; default is: ON
- SAVE\_RLE=<ON|OFF>
  - ; allows to apply RLE compression to the
  - ; PS output
  - ; default is: OFF
- SAVE\_DPI=<value>
  - ; DPI value to use for the output
  - ; default is: 300
- SAVE\_WIDTH=<value>
  - ; Width of the PS page
  - ; default is: 612
- SAVE\_HEIGHT=<value>
  - ; Width of the PS page
  - ; default is: 762

## History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- now uses/requires semaphore system of superview.library V14+
- now correctly handles SVOCheckFile

V3.4 (07.06.1996) :

- added new, rock-solid startup code
- fixed bug in initialization code
- now optionally allows extraction of Postscript (TM) part of EPS files, which requires Ghostscript and GhostScript being correctly installed and configured
- now is able to WRITE EPS files from 24 bit sources: no trailer, but with Postscript (TM) content

\*\*\*\*\*  
 \* Revison history entries before V3.4 have been deleted for space reasons \*  
 \*\*\*\*\*

## 1.50 GhostScript.svobject

© 1996 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

#### Description

~~~~~

GhostScript.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support and PNG.svobject to be present.

It supports importing Postscript (TM) language and PDF files as bitmapped graphics. This is done by using Ghostscript for conversion, giving a DPI resolution and colordepth for the only supported output file format PNG.

GhostScript.svobject will create a temporary file, which then will be parsed through superview.library again.

(See documentation of PNG.svobject for which PNG derivatives currently are supported).

#### Reading :

(Any Postscript (TM) and PDF files as long as readable for Ghostscript).

#### Configuration

~~~~~

This Ghostscript support module has been tested with the Ghostscript 3.53 port from AmiNet, with amiga driver & no ixemul.library: Aladdin Ghostscript is Copyright (C) 1989, 1995 Aladdin Enterprises. All rights reserved. (Ghostscript V4.0 should work as well.)

At first, Ghostscript must have successfully and working been installed (test it via command line access).

For enabling Ghostscript support, you either have to supply an AmigaDOS search path to a directory, where the executable "gs000" is stored (default), or explicitly specify path and/or name of the program by the controlpad entry GS\_PATH (see below).

#### Setting-up Ghostscript

~~~~~

Since the documentation for Ghostscript 3.53 is anything-but-not-very-verbose I decided, to shortly explain here, what steps to perform to successfully install Ghostscript on your system:

- unpack Ghostscript to where you wish it to be located at (e.g. "Work:Ghostscript/")
- add an assign called "Ghostscript:" to your S:User-Startup, which points to that directory (e.g. "Assign Ghostscript: Work:Ghostscript")
- create a directory Ghostscript:fonts (e.g. type "Makedir Ghostscript:fonts" in a shell)
- create four environment variables at both locations,

ENV: and ENVARC:

Variable	Content
GS_DEVICE	amiga
GS_FONTPATH	Ghostscript:fonts
GS_LIB	Ghostscript:data
GS_OPTIONS	

Respect the case, and please note, that the GS\_OPTIONS variable actually is empty, but should be present.

You should use "SetEnv ENV:var content"  
and "SetEnv ENVARC:var content" commands for  
creating these variables.

Please note, that there are many different ways of installing Ghostscript (an assign is not actually needed, and no one actually must supply a fonts directory), but this way it definitely should \*work\*

#### Remarks

~~~~~

- Ghostscript 3.53 has - according to Joop van de Wege - a bug in correctly handling the GS\_FONTPATH variable. Whenever GhostScript.svobject fails in handling a Postscript (TM) file because of missing fonts, this is perhaps the reason. You have to upgrade to Ghostscript V4.0, then.
- please note, that setting DPI to higher values does not necessarily increase image quality, but memory usage (default is DPI=72). Same is to mention for the output color depth: 256 colors give better results, when you've to perform dithering.
- files to be recognized must either have the standard Postscript (TM) header (containing '%!PS-Adobe' in it) or have the file extension ".ps" and begin with '%' (containing instructions for GhostScript).
- PDF files are also supported (extension ".pdf" and beginning with '%'), but having the correct and needed fonts installed is more critical for these (GS\_FONTPATH variable of Ghostscript). Had not enough fonts to successfully test loading of any of these.
- error output of Ghostscript still is directed to stdio/Output(), so problems may be visible transparently

Where to DOWNLOAD from

~~~~~

AmiNet:gfx/show/gs#.LHA (#?=version, CPU and archive content)

#### ControlPad-Switches

~~~~~

```
ControlPad-Name      : "ENV:SuperView-Library/GhostScript.controlpad"
ControlPad-Commands : - GS_PATH=<gs command path plus name>
                        ; how Ghostscript is to be called
                        ; e.g. GS_PATH=Ghostscript:gs000
                        ; default is: gs000
                      - OUTPUTMODE=<PNG256|PNG24BIT>
                        ; Output file format to be used
                        ; (-sDEVICE parameter of Ghostscript's gs).
```

```

; Only PNG supported yet.
; default is: PNG256
- DPI=<value>
; DPI value to use for the converted output
; (-r parameter of Ghostscript's gs)
; default is: 72
- STATUS=<ENABLED|DISABLED>
; allows to disable this module - for example
; to be able to use an other, program-specific
; import-module for the same file format

```

## History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V3.3 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- now correctly handles SVOCheckFile

V3.2 (23.07.1996) :

- added possibility to disable this module, if there's an other, program-specific svobject you would like to use instead (post.library ?)
- fixed error return mechanism for SVO\_Show (accidentally correct ?)

V3.1 (07.06.1996) :

- first version

## 1.51 Targa.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.4

Release Date : 28.09.1996

### Description

~~~~~

Targa.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support.

It supports reading and writing of TGA (Truevision Targa) files.

In detail these are :

Reading :

| Colors | Depth | Organisation               | RLE-Compression |
|--------|-------|----------------------------|-----------------|
| 1      | 1     | planar monochrome          | supported       |
| 256    | 8     | chunky pixel (colors/gray) | supported       |

|          |    |                             |           |
|----------|----|-----------------------------|-----------|
| 32768    | 16 | "HighColor 15/16 Bit" Pixel | supported |
| (24 bit) |    | BGR-Pixel                   | supported |

Writing :

Depending on the Colordepth of the source the following is written :

| Source Colors | Type                      | Destination Colors |
|---------------|---------------------------|--------------------|
| 2 .. 256      | Uncompressed chunky pixel | 256                |
| (24 bit)      | Uncompressed BGR          | (24 bit)           |

Remarks

~~~~~

- 32 Bit graphics files are not supported yet.
- ColorMaps have to be of type "3-Byte BGR".
- The flags for "mirroring" Images vertically and/or horizontally are not fully interpreted yet, nevertheless these will be reported by SVL\_FileInfoRequest().  
If the VERTINV flag is not set, the picture will be assumed to be written as "from bottom to top", otherwise as "from top to bottom". Some programs do not set these flags right, when writing, so that you might get just the opposite result as expected.  
The HORIZINV flag is currently ignored: when reading such a picture as usual, you'd get a mirrored image. But this flag is also set wrong sometimes ...

History

~~~~~

V4.4 (28.09.1996) :

- fixed bug in file recognition  
(-> Jürgen Schäfer)

V4.3 (04.09.1996) :

- more restrictive file recognition  
(-> Jürgen Schäfer)

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.3 (07.06.1996) :

- added new, rock-solid startup code
- fixed bug in initialization code

\*\*\*\*\*  
\* Revision history entries before V3.3 have been deleted for space reasons \*

\*\*\*\*\*

## 1.52 WPG.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

Description

~~~~~

WPG.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support.

It supports reading of WPG (WordPerfect) BitMap files.

In detail these are :

Reading :

- WPG Bitmap (greyscaled) with 1, 2, 4 or 8 bits colordepth
- WPG Bitmap (colorMap) with 1, 2, 4 or 8 bits colordepth

Remarks

~~~~~

WordPerfect WPG files do not necessarily have to contain bitMap graphics, they also may contain various other data, e.g. vector graphics.

If a WPG file contains a bitMap graphic in any of its chunks, it will be loaded, otherwise the file will be rejected.

If a file does not contain any color information, WPG.svobject will generate greyscales by default.

This will be mentioned in the file-info requester.

This version actually has been tested with graphics with 1, 4 and 8 Bit colordepth (2, 16 and 256 Colors).

Due to the fact, that the 2 bit-routine is identically to the 1 bit-routine you should not get any problems with those pictures.

History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.3 (09.06.1996) :

- added new, rock-solid startup code

```
*****
* Revison history entries before V3.3 have been deleted for space reasons *
*****
```

## 1.53 SunRaster.svobject

© 1994-96 by Andreas R. Kleinert.

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

### Description

~~~~~

SunRaster.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support.

It supports reading and writing of SunRaster (RAS) files.  
In detail these are :

### Reading :

- SunRaster with 2 Colors ( 1 Bit, planar)
- SunRaster with 256 Colors ( 8 Bit, chunky pixel)
- SunRaster with 24 bit (24 bit, R-G-B)

### Writing :

Depending on the Colordepth of the source the following is written :

| Source Colors | Type                      | Destination Colors |
|---------------|---------------------------|--------------------|
| 2..256        | Uncompressed Chunky Pixel | 256                |
| (24 bit)      | Uncompressed 24 bit RGB   | (24 bit)           |

### Remarks

~~~~~

- files with Colorbits other than 1, 8 or 24 are not supported yet
- only RGB-planar colormaps supported (or monochrome, without map)
- max. 1 plane of bitmap data allowed

### History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set

V3.3 (07.06.1996) :



- added new, rock-solid startup code
- fixed bug in initialization code

```
*****
* Revison history entries before V3.3 have been deleted for space reasons *
*****
```

## 1.54 SGI.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

### Description

~~~~~

SGI.svobject is an external library module for the superview.library, which needs any SVDriver with 24 bit Oneplane-Support.

It supports reading and writing of SGI files.

In detail these are :

Colors Depth	Organisation	RLE-Compression
256 8	8 Bit Chunky Pixel (gray)	supported
(24 Bit)	8:8:8 24 bit RGB	supported

Writing :

Source Colors	Type	Destination Colors
(24 bit)	Uncompressed RGB	(24 bit)

### Remarks

~~~~~

- Currently files with BPC-Values of 2 are not supported, which means that pixel data has to be stored in bytes, not words (either gray 8 Bit or 24 bit RGB as 8:8:8, not 16 or 16:16:16)

### History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correcly handles SVOCheckFile

V3.3 (07.06.1996) :

- added new, rock-solid startup code
- fixed bug in initialization code

```
*****
* Revison history entries before V3.3 have been deleted for space reasons *
*****
```

## 1.55 PICT.svobject

© 1995-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

### Description

~~~~~

PICT.svobject is an external library module for the superview.library, which needs any SVDriver with 24 bit Oneplane-Support.

It supports reading of Mac PICT-2 Metafile graphics.  
In detail these are :

### Reading :

- Mac PICT-2 with 1, 2, 4, 8 or 24 bit ColorDepth  
always as 24 bit RGB Data (JPEG optionally as dithered 8 Bit)

### Remarks

~~~~~

- the file extension has to be ".pct" (as on PCs), ".pic" or ".pict", otherwise it will be rejected  
(there do more checks take place, but those are less relieable)
- pictures always will be exported as 24 bit RGB data
- font handling always will operate with the internal standard font instead of the appropriate ones  
(derived from PBM package, see Credits)

### Limitations

~~~~~

Please note, that the Decoding Routines of the Library are not (yet) fully re-entrant, so that only one Task may Decode a picture at a time. The Library itself manages it, that the other Task has to Delay() until the Decoding Routines are "free" again.  
Since Decoding usually is managed very fast, this should not actually matter.

### History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.5 (21.05.1996) :

- fixed bug, which may have caused upto 8 bytes of the library (base) to possibly be overwritten

```
*****
* Revision history entries before V3.5 have been deleted for space reasons *
*****
```

## 1.56 Pictor.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

Description  
~~~~~

Pictor.svobject is an external library module for the superview.library, which needs any SVDriver with Bit-/Oneplane-Support.

It supports reading of Pictor/PC Paint (PIC) files.  
In detail these are :

Reading :

- Files with 1, 4, or 8 Bit ColorDepth (monochrome or with EGA or VGA palette).

History  
~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.3 (09.06.1996) :

- added new, rock-solid startup code

```
*****
```

---

\* Revision history entries before V3.3 have been deleted for space reasons \*  
\*\*\*\*\*

## 1.57 MAC.svobject

© 1994-96 by Andreas R. Kleinert.

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

Description  
~~~~~

MAC.svobject is an external library module for the superview.library,  
which needs any SVDriver with Bitplane-Support.

It supports reading of MAC (MacIntosh MacPaint) files.  
In detail these are :

Reading :

- MAC Black & White 576x720

Remarks  
~~~~~

MacPaint files, which are to be loaded into "MAC.svobject" should contain  
the specific MacBinary Header (first the 128 Byte-Header, then the  
MacPaint specific data appended to it).

In the PC area, there may sometimes files be found, which just contain  
the MacPaint 576x720 black and white Data.

These files - without the header - can only be identified via the three  
leading zero bytes at the beginning of the 512 Byte MacPaint header.  
But there's never a 100% guarantee that a File with three leading zeroes  
really is a MacPaint File, so we also request a filename ending with  
".mac" or ".MAC" in this special case.

History  
~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.3 (09.06.1996) :

- added new, rock-solid startup code

\*\*\*\*\*  
\* Revision history entries before V4.1 have been deleted for space reasons \*

\*\*\*\*\*

## 1.58 JPEG.svobject

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

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

Version : 4.3

Release Date : 28.09.1996

Description

~~~~~

JPEG.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support.

It supports reading and writing of JPEG files as written by "The Independent JPEG Group's JPEG Software" (release 6).

In detail these are :

Reading :

- JPEG (IJG-JFIF), with output in 256 Colors or 24 bit

Writing :

- JPEG (IJG-JFIF), from upto 8 or 24 bit Input

Remarks

~~~~~

\* VMEM:

- \* Temporary data might be written to a directory assigned to "VMEM:", if necessary, but please note, that this might affect your HardDisk, if this program crashes (e.g. on a corrupt JPEG-picture) !  
So don't blame me, if this happens, but select a safe place for this temporary-file directory !

\* Writing JPEG-files :

- \* Note, that JPEG-compression is lossy, which means that the original picture cannot be reconstructed totally.  
Only JPEG-compress files, of which you have backups, or files which you never want to edit and enhance again.  
You should always control the results from saving an JPEG-File : in some special cases (e.g. many thin lines on the screen) there may occur strong differences to the original picture.

ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/JPEG.controlpad"

ControlPad-Commands :  
; ; STANDARD settings  
; \*\*\*\*\*

```
- COLORDEPTH=<8|24>
; For READING colored 24 bit images:
; should the output be done as 8 or 24 bit Data ?
; default is 8 Bit data. Grayscaled graphics
; will _always_ be exported as 8 Bit
- QUANTIZATION=<0..100>
; For WRITING 8/24 bit images as 24 bit JPEG:
; 0 - poor quality, big compression
; 100 - best quality, no compression
; never use such extreme values, but try
; something like 20, 50, 75
; default is: 75)
;
; ADVANCED settings
; *****
- FORCE_GRAY_DECODE
; For READING images:
; this switch overrides the selected depth
; for colored images and always exports them
; as 8 Bit grayscaled pictures
; default is: not set
- DECODE_METHOD=<FAST_INTEGER|SLOW_INTEGER|
                FLOATING_POINT|FASTEST_INTEGER>
; For READING images:
; how to DECODE the 24 bit or gray data
; default is: FAST_INTEGER
- DITHERMODE=<DITHER_FLOYD-STEINBERG|
                NO_DITHERING|DITHER_ORDERED>
; For READING and dithering 24 bit images
; to 8 Bit images:
; how and whether to DITHER the 24 bit data
; default is: DITHER_FLOYD-STEINBERG
- COLOR_QUANTIZING=<FAST|SLOW>
; For READING and dithering 24 bit images
; to 8 Bit images:
; whether to QUANTIZE the 24 bit data within
; two passes
; default is: SLOW
- UPSAMPLING=<ON|OFF>
; For READING and dithering 24 bit images
; to 8 Bit images:
; whether to do fancy upsampling on the 24 bit
; default is: ON
- FORCE_GRAY_ENCODE
; For WRITING images:
; this switch causes grayscales to be written
; no matter, whether the input was colored
; default is: not set
- ENCODE_METHOD=<FAST_INTEGER|SLOW_INTEGER|
                FLOATING_POINT|FASTEST_INTEGER>
; For WRITING images:
; how to ENCODE the 24 bit or gray data
; default is: FAST_INTEGER
- OPTIMIZE=<ON|OFF>
; For WRITING 24 bit images:
; whether to optimize the generated huffman
; code (good compression, but slow)
```

---

```

; default is: OFF
- PROGRESSIVE=<ON|OFF>
; For WRITING 24 bit images:
; whether to write progressive JPEG files
; default is: OFF

```

#### History

~~~~~

V4.3 (28.09.1996) :

- upgraded to V6a sources, since some people seem to be so "version fanatic", that it doesn't matter, whether an upgrade actually makes any sense for our uses %-)

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (08.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile
- 68030 version of JPEG.svobject had been linked with 68000 version of JFIF library. Fixed (not a bug, bug simply a nearly a no-op).
- fixed Enforcer hit in the Amiga-specific memory-handler initialization routine, which popped up when trying to load/save a JPEG file.

```

*****
* Revison history entries before V4.1 have been deleted for space reasons *
*****

```

## 1.59 PCD.svobject

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

#### Description

~~~~~

PCD.svobject is an external library module for the superview.library, which needs any SVDriver with 24 bit Oneplane-Support.

It supports reading of the unpacked resolutions of PCD-files, as BASE/16 (192x128), BASE/4 (384x256) and BASE (768x512).

It DOES NOT READ the BASE\*4 (1536x1024) and BASE\*16 (3072x2048) resolutions, since these are encrypted in a way, which is not only undocumented, but also copyrighted by Kodak.

Legal Issues

~~~~~

This program module is FREEWARE.

The author does not claim any copyrights on the code, which is used to decode the YUV-data fromout the PCD-file (see credits) or on the other PCD-related information.

Copyright is only claimed for the program as a whole, which means that some parts of the library module, which are also used within other SVOobjects, are copyrighted by the author.

If the writing or the publication of this program should ever be considered to be partly any kind of a violation of third party copyrights, it is hereby expressed that the usage of this program is only allowed, if any user of it agrees to the following:  
If the case as described above takes place, any user has to delete any copies of this program immediately when he gets informed about it.

ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/PCD.controlpad"

ControlPad-Commands : - OUTPUTFORMAT=<BASE/16 | BASE/4 | BASE >  
; specifies the output resolution to be used

History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- new GST
- now correctly handles SVOCheckFile

V3.3 (07.06.1996) :

- added new, rock-solid startup code
- fixed bug in initialization code

\*\*\*\*\*  
\* Revison history entries before V3.3 have been deleted for space reasons \*  
\*\*\*\*\*

## 1.60 FastILBM24.svobject

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

Description



~~~~~

FastILBM24.svobject is an external library module for the superview.library, which needs any SVDriver with Oneplane-Support.

It has especially been written to allow fast and "low-memory" loading and dithering of ILBM 24 bit graphics.

This means, that no compressed or uncompressed 24 bit data will be completely loaded into memory, but directly be dithered to HAM6/8, so that even people with less memory and no graphics card will be able to view those nice 24 bit graphics.

May be deactivated, even if installed; may be switched between HAM6 and HAM8; allows increasing/decreasing of internal cache buffer.

ControlPad-Switches

~~~~~

```
ControlPad-Name      : "ENV:SuperView-Library/FastILBM24.controlpad"
ControlPad-Commands : - BUFFERSIZE=<Size in Bytes>
                        ; how many bytes should be buffered
                        (more = faster)
                        ; Default is 8192, minimum is 256
- PASSTHROUGH
  ; do not recognize ANY pictures, so that
  ; they e.g. might be passed through to
  ; ILBM.svobject instead
- DITHERMODE=<HAM6_QUICK|HAM8_QUICK>
  ; Default is HAM6_QUICK
```

History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- new GST
- now correctly handles SVOCheckFile

V3.3 (07.06.1996) :

- added new, rock-solid startup code
- fixed bug in initialization code

```
*****
* Revison history entries before V3.3 have been deleted for space reasons *
*****
```

## 1.61 YUVN.svobject

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 18.08.1996

#### Description

~~~~~

YUVN.svobject is an external library module for the superview.library, which needs any SVDriver with 24 bit Oneplane-Support.

It supports reading and writing of IFF-YUVN (YUVN) files.

In detail these are :

#### Reading :

- IFF-YUVN 24 bit (gray, 411, 422, 444, 211, 222)

Gray will be imported as 24 bit with R=G=B.

Use "ExtractGrayScales" operator to reduce memory usage - and redundancy

#### Writing :

- IFF-YUVN 24 bit (411)

Reading/writing from/to ClipBoard is supported.

#### Remarks

~~~~~

- only 24 bit sources will be written as IFF-YUVN.  
256 color graphics are NOT automatically transformed to 24 bit, so that you may have to use the "AnyTo24Bit" operator before saving.

#### History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.4 (08.06.1996) :

- fixed bug in initialization code
- fixed chunk recognition (may have read beyond ID array size)
- changed Clipboard file recognition

\*\*\*\*\*  
\* Revision history entries before V3.4 have been deleted for space reasons \*

\*\*\*\*\*

## 1.62 DEEP.svobject

© 1995-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2

Release Date : 29.08.1996

Description

~~~~~

DEEP.svobject is an external library module for the superview.library, which needs any SVDriver with 24 bit Oneplane-Support.

It supports reading and writing of specific IFF-DEEP (DEEP) files.

In detail these are :

Reading :

- IFF-DEEP 24 bit RGB 8:8:8, uncompressed  
and 32 bit RGBA 8:8:8:8, uncompressed (Alpha channel ignored)
- IFF-DEEP 24 bit RGB 8:8:8, RLE-compressed  
and 32 bit RGBA 8:8:8:8, RLE-compressed (Alpha channel ignored)

Writing :

- IFF-DEEP 24 bit RGB 8:8:8, uncompressed

Reading/writing from/to ClipBoard is supported.

Remarks

~~~~~

- only 24 bit sources will be written as IFF-DEEP.  
There's no conversion done from e.g. 256 Colors to 24 bit to perform the requirements of this file format.
- other compression methods (e.g. Huffman) aren't yet supported due to a lack of information on implementation

History

~~~~~

V4.2 (29.08.1996) :

- now has new disk read/write flags set
- now supports reading of 32 Bit RGB pictures with Alpha channel as well (Alpha channel is ignored)
- now checks for correct order of RGB (or RGBA)

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.3 (08.06.1996) :

- added new, rock-solid startup code
- fixed bug in initialization code
- fixed chunk recognition (may have read beyond ID array size)
- fixed Clipboard file recognition (DGBL had to be first chunk)

```
*****
* Revison history entries before V3.3 have been deleted for space reasons *
*****
```

## 1.63 QRT.svobject

© 1995-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

Description  
~~~~~

QRT.svobject is an external library module for the superview.library, which needs any SVDriver with 24 bit Oneplane-Support.

It supports reading and writing of QRT (POV RayTracer) files.  
In detail these are :

Reading :

- QRT Dump 24 bit

Writing :

- QRT Dump 24 bit

Remarks  
~~~~~

- only 24 bit sources will be written as QRT.  
There's no conversion done from e.g. 256 Colors to 24 bit to perform the requirements of this file format.

History  
~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- made sure, that always correct svgfx\_Version is set
- now correcly handles SVOCheckFile

V3.5 (07.06.1996) :

- fixed bug in initialization code

```
*****
* Revison history entries before V3.5 have been deleted for space reasons *
*****
```

## 1.64 C-Source.svobject

© 1996 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.2  
Release Date : 18.08.1996

Description

~~~~~

C-Source.svobject is an external library module for the superview.library.

It creates C-Source code in chunky/RGB notation from any input graphics  
(≤ 8 Bit and HAM as chunky, 24 bit as RGB).

In detail these are :

Writing :

- 8 Bit as C-Source with UBYTE [] pixel array and UBYTE [][][3] colormap  
(HAM6/8 indicated by flag #define)
- 24 bit as C-Source with UBYTE [][][3] pixel array (RGB data)

For simple extraction of a colormap from a graphics file, you should  
apply the Crop operator with parameters

```
CROP_LEFTEDGE=0
CROP_TOPEDGE=0
CROP_WIDTH=1
CROP_HEIGHT=1
```

and then save the result as C-Source.

Remarks

~~~~~

- this module uses buffered I/O, but nevertheless is quite slow for  
larger graphics (I/O buffer is set to 64K for OS version ≥ 39)
- usage of RAM-Disk for output buffers is strongly suggested
- the module does raw output, which should be compileable with  
any Amiga C compiler - and any other compiler, when simply  
replacing

```
#include <exec/types.h>
```

with

```
typedef unsigned char UBYTE;
typedef unsigned long ULONG;
```

- the "endmark" entries at the end of any written array can simply be removed; they actually have no meaning, but simplify the process of writing the data to the file

#### History

~~~~~

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- now correctly handles SVOCheckFile

V3.2 (07.06.1996) :

- fixed bug in initialization code

V3.1 (16.04.1996) :

- first version: yet only with chunky/RGB support and not very fast

## 1.65 UtahRLE.svobject

This one is not included with this distribution, but can be found on AmiNet or any related BBS.

Look out for an archive called "svoUtah34.LHA" or similar.

## 1.66 CyberDataType.svobject

© 1996 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 4.3

Release Date : 13.09.1996

#### Description

~~~~~

CyberDataType.svobject supports (CyberGraphX') picture.datatype V43, which upgrades DataTypes to 24 bit handling.

Ralph Schmidt's picture.datatype V43 does allow loading of 24 bit pictures by enhancing the datatype system with additional functionality. To be able using these features with SuperView, you either need a registered CyberGraphX installation or the freely available cyberncg.library to be used together with CyberDataType.

When CyberGraphX (or cyberncg.library) and picture.datatype V43+ are both installed in your system, this module will supersede the internal DataType svobject (for graphics only) and thus allow extraction of 8 Bit

Chunky and 24 bit RGB graphics from Picture DataTypes.

You need special DataTypes for that (as coming with picture.datatype V43).

Reading :

- any DataTypes of class "picture" in either 8 Bit (standard)
- and/or 8 and 24 bit  
(with special datatype modules for picture.datatype V43)

Remarks

~~~~~

- This modules does for example not handle animations or other datatypes, which will still be passed through the standard datatype handler, when the library is configured to handle "ANYDATATYPES".
- Please note, that this module will install itself already, if cybergraphics.library or cyberncg.library are present, but it will only start working, when also picture.datatype V43 is available. Otherwise the default datatypes handling will take place.

History

~~~~~

V4.3 (13.09.1996) :

- now opens cybergraphics and cyberncg-libs just when needed, not before, also datatypes
- removed checking for cyberncg-lib (or cybergraphics, thus will be loaded always now

V4.2 (18.08.1996) :

- now has new disk read/write flags set

V4.1 (06.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)
- did never set svgfx\_Version (was 0)
- made sure, that always correct svgfx\_Version is set
- now correctly handles SVOCheckFile

V3.2 (24.07.1996) :

- now also works with cyberncg.library and without CyberGraphX (if cybergraphics.library or cyberng.library are installed, but no picture.datatype V43, it will be loaded nevertheless, but doesn't do any harm to the original datatypes handler)
- could have crashed at startup-time, when picture.datatype V43 not already had been loaded (Ramlib-process problem)
- now compiled for 68000, since some people may wish to use picture.datatype V43 and cyberncg.library on these systems (CyberGraphX is for >= 68020 only, but did not find such a note for the rest)

V3.1 (09.02.1996) :

- first version (tested with picture.datatype V43.667)
-

## 1.67 ECS.svdriver

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 2.9

Release Date : 01.08.1996

### Requirements

~~~~~

- OS V2.04+ (V37)+ and its libraries
- superviewsupport.library V6+
- AMIGA with Enhanced Chip Set (ECS)

### Description

~~~~~

ECS.svdriver is an external SV\_GfxBuffer display module for superview.library, which allows object-oriented displaying on ECS displays.

This Driver supports the following :

| Dimensions | Depth  | Type                    |
|------------|--------|-------------------------|
| [ECS]      | [ECS]  | BITPLANE                |
| [ECS]      | 8/(24) | ONEPLANE (Chunky Pixel) |

The 8-Bit mode will perhaps only work on ECS systems which have any Graphic Card installed, which allows 256 or more colors in a way of an Intuition emulation.

24 bit Graphics will be displayed as "best guess of 256 colors" if no SVOperator is specified.

Autoscrolling of Screens larger than the actual display is supported : Just move the mouse to the boundings !

### ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/ECS.controlpad"

ControlPad-Commands : - 24BITOPERATOR=<OperatorName>  
     ; (case-sensitive, ".svoperator" may be added)  
     ; e.g. "24BITOPERATOR=24BitToHAM.svoperator"  
     ; or "24BITOPERATOR=ExtractGrayScales"  
     ; specifies, which operation should be performed  
     ; on 24 bit graphics before displaying them  
     ; (if not specified displaying will fail)  
   - 8BITOPERATOR=<OperatorName>  
     ; (case-sensitive, ".svoperator" may be added)  
     ; e.g. "8BITOPERATOR=ExtractGrayScales"  
     ; specifies, which operation should be performed  
     ; on non-ECS graphics (more than 16 Colors in  
     ; HighRes, more than 32 Colors in LowRes, HAM8)  
     ; (if not specified, will be tried to display)  
   - BITMAPCOPY=<DIRECT|RTG>  
     ; "BITMAPCOPY=RTG" prevents ECS.svdriver from



```

; directly copying into Bitmaps, which will
; result in a usage of more memory, but keeps it
; working.
; Default is "BITMAPCOPY=DIRECT".
- SCREENINFRONT
; Put Screen to front _before_ the graphics
; has been displayed (useful with GfxCards)

```

#### History

~~~~~

V2.9 (01.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library

V2.8 (21.12.1995) :

- added REV. 2 of new, rock-solid startup code
- fixed several possible crashing reasons
- tried to use LAYERS\_NOBACKFILL with SA\_BackFill and WA\_BackFill even under V37-V38. This caused an odd-address error since the given dummy-pointer to address 1 is only valid for V39 or greater. With V2.7 of ECS.svdriver displaying under V38 and below perhaps was impossible. Fixed.  
(-> reported by Klaus Schneider and Sven Drieling)

V2.7 (29.10.1995) :

- faster closing
- recompiled with SAS/C V6.56

```

*****
* Revison history entries before V2.7 have been deleted for space reasons *
*****

```

## 1.68 AGA.svdriver

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 2.9  
Release Date : 01.08.1996

#### Requirements

~~~~~

- OS V3.00+ (V39)+ and its libraries (takes advantage of V40)
- superviewsupport.library V6+
- AMIGA with AGA ChipSet (ECS still supported, but with restrictions) or an appropriate Graphics Card with Workbench Emulation

#### Description

~~~~~

AGA.svdriver is an external SV\_GfxBuffer display module for superview.library, which allows object-oriented displaying on

AGA displays.

This Driver supports the following :

| Dimensions | Depth  | Type                    |
|------------|--------|-------------------------|
| [AGA]      | [AGA]  | BITPLANE                |
| [AGA]      | 8/(24) | ONEPLANE (Chunky Pixel) |

24 bit Graphics will be displayed as "best guess of 256 colors" if no SVOperator is specified.

Autoscrolling of Screens larger than the actual display is supported : Just move the mouse to the boundings !

ControlPad-Switches

~~~~~

```
ControlPad-Name      : "ENV:SuperView-Library/AGA.controlpad"
ControlPad-Commands : - 24BITOPERATOR=<OperatorName>
                        ; (case-sensitive, ".svoperator" may be added)
                        ; e.g. "24BITOPERATOR=24BitToHAM.svoperator"
                        ; or   "24BITOPERATOR=ExtractGrayScales"
                        ; specifies, which operation should be performed
                        ; on 24 bit graphics before displaying them
                        ; (if not specified, "best guess" colors will
                        ; be used, which is really slow)
- BITMAPCOPY=<DIRECT|RTG>
  ; "BITMAPCOPY=RTG" prevents AGA.svdriver from
  ; directly copying into Bitmaps, which will
  ; result in a usage of more memory, but keeps it
  ; working.
  ; Default is "BITMAPCOPY=DIRECT".
- SCREENINFRONT
  ; Put Screen to front before the graphics
  ; has been displayed (useful with GfxCards)
```

History

~~~~~

V2.9 (01.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library

V2.8 (21.12.1995) :

- added REV. 2 of new, rock-solid startup code
- fixed several possible crashing reasons

V2.7 (29.10.1995) :

- faster closing
- recompiled with SAS/C V6.56

\*\*\*\*\*  
 \* Revision history entries before V2.7 have been deleted for space reasons \*

\*\*\*\*\*

## 1.69 CyberGraphics.svdriver

© 1995-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 2.9

Release Date : 01.08.1996

### Requirements

~~~~~

- OS V3.00+ (V39)+ and its libraries
- 68020 or greater processor
- superviewsupport.library V6+
- cybergraphics.library V40+
- AMIGA with CyberGraphX System and appropriate Graphics Card

### Description

~~~~~

CyberGraphics.svdriver is an external SV\_GfxBuffer display module for superview.library, which allows object-oriented displaying on Amigas with the 16/24 bit CyberGraphX Workbench Emulation System.

This Driver displays any 1..8 or 24 bit graphics via the CyberGraphX System. Graphics upto 256 Colors will be displayed via usual graphics routines, 24 bit graphics will be displayed via CyberGraphX in either 16 or 24 bit (16 Bit is default, because the unregistered version of CyberGraphX does not support more and I guess, that not yet all the users out there registered it already).

HAM6/8 will be converted to 24 bit, which then will either be displayed as 16 or 24 bit.

The driver will not open any Screens in standard ECS/AGA modes, since it makes more sense to use AGA.svdriver in such a case. If no CyberGraphX screenmode is specified, it will be generated, which is just the same effect like with an intelligent screen promoter. No ECS/AGA screenmodes will be passed through, since this is neither useful nor practicable on two Monitor systems or with single 31kHz-limited Monitors.

### Credits

~~~~~

Thanks to Ingenieurbüro Helfrich, for supplying the PiccoloSD64 card.

The CyberGraphX Software is of course copyrighted by its authors, which is hereby expressively respected in all points.

---

## ControlPad-Switches

~~~~~

```
ControlPad-Name      : "ENV:SuperView-Library/Cybergraphics.controlpad"
ControlPad-Commands : - EMUSCREENDEPTH=<16|24>
                      ; beginning depth for opening CyberGraphX Screens
                      ; Default is 16 Bit (which will also be tried
                      ; when opening of a 24 bit Screen fails)
                      ; This does not concern colordepths < 16 Bit,
                      ; except HAM6/8.
                      - SMALLSCREENS
                      ; unless this KeyWord is specified, it is not
                      ; tried to open screens smaller than 320x240
```

## History

~~~~~

V2.9 (01.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library

V2.8 (10.02.1996) :

- new C startup-code was compiled for 68020. Fixed.
- this revision history was slightly wrong

V2.7 (13.01.1996) :

- added REV. 2 of new, rock-solid startup code
- using new GST
- now requesting superviewsupport.library V6+
- did distort any 8 Bit odd width graphics (not divideable by 16 without rest) when displaying under OS V3.00 (V39). This did not happen with graphics < 320x240 and the SMALLSCREENS option unset and it did not concern OS V3.1 (V40) users. (-> Jürgen Schäfer)
- May now be slightly slower in displaying such graphics, since these will be displayed line-wise. Aligned graphics will be displayed using the old method.

V2.6 (29.10.1995) :

- faster closing
- recompiled with SAS/C V6.56

```
*****
* Revision history entries before V2.6 have been deleted for space reasons *
*****
```

## 1.70 EGS7.svdriver

© 1994-96 by Andreas R. Kleinert.

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 2.4

Release Date : 01.08.1996

## Requirements

~~~~~

- OS V2.04+ (V37)+ and its libraries
- superviewsupport.library V6+
- egs.library V1+ (should be V7+)
- egsintui.library V1+ (should be V7+)
- egsgfx.library V1+ (should be V7+)
- AMIGA with EGS Graphic Card (or EGS distribution with Amiga Emulation)

## Description

~~~~~

EGS7.svdriver is an external SV\_GfxBuffer display module for superview.library, which allows object-oriented displaying on EGS V7 (Enhanced Graphic System).

This Driver displays any 1..8 or 24 bit graphics via the EGS7 System. Any source Graphics with less than 24 bit, like 256 Color or HAM6/8 graphics, are converted to 24 bit before displaying them.

This EGS7.svdriver is based on my former "EGS.svdriver", which is to be replaced by the new one. But of course you may still continue using the older one.

The difference is, that the EGS7.svdriver really does display any of the available source graphics formats (even HAM) and is x-times faster than the old one. Also, it is now possible to directly close the EGS-Displaywindow via its Close-Gadget, since some kind of self-written Intuition-Emulation enables us to do so.

## Credits

~~~~~

Thanks to Ingenieurbüro Helfrich, for supplying the PiccoloSD64 card.

The EGS7 Software is of course copyrighted by its authors, which is hereby expressively respected in all points.

## Notes

~~~~~

This EGS-SVDriver should still work with the AMIGA driver for EGS.

When using the ECS/AGA emulation, you should set the max. possible ColorDepth in the ScreenMode preferences program.

## AGA:

With AGA any Graphics with less than 256 Colors should be displayed 100% correctly. Only 256 Color-graphics will usually have some Colors wrong, because those are obtained by the EGS-System for the Display itself (Window-Borders, etc.), so that they usually can't be used for the graphics. 24 bit graphics will be dithered to 256 Colors under AGA (usually GreyScaled).

## ECS:

Using the ECS emulation will perhaps nearly always result in very ugly Colors, if you're displaying more than, let's say, 8 Colors. This results out of the maximum ColorDepth of 16 Colors in Hires,

of which some - see AGA notes - are already reserved.  
Of course 24 bit graphics may also be dithered to 16 Colors/GreyScales,  
but better don't try it out ...

#### History

~~~~~

V2.4 (01.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using new startup-code now (oops, forgot that before)
- new GST
- using utility.library
- smaller now
- and more

V2.3 (29.10.1995) :

- recompiled with SAS/C V6.56

V2.2 (9.10.1995) :

- fixed version counting for info structure

V2.1 (28.7.1995) :

- rewrote "EGS.svdriver" and called it "EGS7.svdriver" now
- is x-times faster now
- displays any available GfxBuffer format (even HAM) in 24 bit now
- uses kind of self-made Intuition Emulation now, no longer a requester-window on the Workbench has to be used to close the EGS-Screen, but instead using the EGS-Window's Close-Gadget suffices now (also the most important IDCMP actions, like IDCMP\_MOUSEBUTTONS, -RAWKEY, -VANILLAKEY and -CLOSEWINDOW are already supported). Suffices e.g. for SuperView.

## 1.71 Picassoll.svdriver

© 1995-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 2.4

Release Date : 01.08.1996

#### Requirements

~~~~~

- OS V2.04+ (V37)+ and its libraries
- superviewsupport.library V6+
- vilintuisup.library V1+
- AMIGA with Picasso II Graphics Card and Software

#### Description

~~~~~

PicassoII.svdriver is an external SV\_GfxBuffer display module for superview.library, which allows object-oriented displaying on

Amigas with the Picasso Workbench Emulation System.

The related Picasso-libraries are needed.

This Driver supports the following :

| Source   | Depth | Special Modes | Dimensions           | Dest. Depth |
|----------|-------|---------------|----------------------|-------------|
| ONEPLANE | 8/24  | HAM6, HAM8    | [ Picasso-Dependent] | 8 / 24      |
| BITPLANE | 1..8  | HAM6, HAM8    | [ Picasso-Dependent] | 8 / 24      |

HAM6/8 will be converted to 24 bit, which then will either be displayed as 16 or 24 bit.

24 bit will either be displayed in 16 or 24 bit.

#### Credits

~~~~~

Thanks to Florian Zeiler (Irseesoft) for supplying the Gfx Card.

The Picasso Emulation Software is of course copyrighted by Village Tronic Marketing GmbH, which is hereby expressively respected in all points.

#### Known Bugs

~~~~~

##### Problem:

As with V1.8 (6. Apr 1994) of the Picasso II Software Set, the vilintuisup.library sometimes supplies Screens, which differ between internal dimensions and actual dimensions.

A 24 bit graphics with 1165x712 would be displayed on a Screen with correct dimensions in best ViewMode. But for some reason, the Intuition Part of the Screen will be opened in 1168x712 with a related ViewMode. Whenever this ViewMode is the best fitting one, this seems to make problems to the Picasso Emulation. If the ViewMode is one with a larger or smaller display (so that either Autoscroll has to be used, or the graphics does not fill the whole screen) this does not occur.

This is not a bug of the PicassoII.svdriver, since width adjustments will occur independently from the viewmode selection.

##### Solution:

Whenever the Picture is weirdly distorted, at first try an other viewmode. This bug does not seem to cause crashes, only destroyed displays.

Maybe a newer version of the Picasso Software already did fix this (seems to be an alignment problem when copying from RAM into the Picasso onboard-memory).

##### Problem:

When using AUTOSCROLLADJUST the Picasso Software may destroy about two lines within the displayed picture (get black).

##### Solution:

Don't use it.

#### ControlPad-Switches

~~~~~

```

ControlPad-Name      : "ENV:SuperView-Library/PicassoII.controlpad"
ControlPad-Commands : - SMALLSCREENS
                      ; unless this KeyWord is specified, it is not
                      ; tried to open screens smaller than 320x240
- BLITTER
  ; uses the Picasso blitter to copy graphics
  ; into screens. Otherwise CPU is used.
  ; On 68000 systems you may wish to use the blitter
- EMUSCREENDEPTH=<16|24>
  ; beginning depth for opening Picasso Screens
  ; Default is 24 bit (if opening fails, it is
  ; also tried to open a 16 Bit Screen, then)
- SCREENMODEREQUEST
  ; if this one is specified, the SVDriver ITSELF
  ; will open a ScreenMode-Requester and ask
  ; for an appropriate Screenmode to use.
  ; Useful, if you always like to change modes.
- AUTOSCROLLADJUST
  ; this keyword will force Autoscroll whenever
  ; it would make sense, but the Picasso Software
  ; would not manage it by itself (when either
  ; only width or height need to be autoscrollled)

```

#### History

~~~~~

V2.4 (01.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using new startup-code now (oops, forgot that before)
- new GST
- using utility.library
- smaller now
- and more

V2.3 (29.10.1995) :

- recompiled with SAS/C V6.56

```

*****
* Revison history entries before V2.3 have been deleted for space reasons *
*****

```

## 1.72 OPAL.svdriver

Just a short note before using and installing OPAL.svdriver  
and (of course ;-) reading the following documentation:

There's a little problem causing some OPAL programs not to be displayed  
correctly on certain systems with specific chipsets.

That was the reason why it did not work on the machines of some  
people - mine was one of these - but nevertheless it is easy to  
workaround and even already has been described within the official  
OpalVision Reference Manual [page 326].



To speak clearly, the color palette has to be adjusted concerning Color 0 (the magic color), but only the BLUE component, within the Palette Preferences.  
Any other colors and components are freely selectable.

OCS and ECS Chipset Machines with Zorro Bus (2000-3000)

BLUE of COLOR 0 must be either 1, 3, 5, 7, 9, 11, 13, or 15.

AGA Chipset Machines with Zorro Bus (4000)

BLUE of COLOR 0 must have Bit 4 set, which restricts it to one of the following ranges:

|            |    |         |    |         |    |         |
|------------|----|---------|----|---------|----|---------|
| 16-31      | or | 48-63   | or | 80-95   | or | 112-127 |
| or 144-159 | or | 176-191 | or | 208-223 | or | 240-255 |

Otherwise you may get a black or weirdly distorted screen (I tested it ;-)  
or must use some tricks and Amiga+M fiddling to get a picture.

\*\*\*\*\*  
included the original documentation at this place  
\*\*\*\*\*

OPAL.svdriver/Documentation

OPAL.svdriver/Documentation

#### PROGRAM

OPAL.svdriver  
Version 2.3  
Release date: 29.6.95

#### COPYRIGHT

© 1995 Paul Huxham and Steve Quartly.  
Bonusware, all rights reserved.

#### DISTRIBUTION

Freely distributable with any non-commercial application that uses superview, however all copyright remains with the authors. This distribution archive may not be split for further distribution.

#### PURPOSE

OPAL.svdriver is an external display driver module for superview.library. It can display superview buffers on an OpalVision display card.

#### CONTENTS

OPAL.svdriver - This is the driver to copy to LIBS:svdrivers.  
OPAL.svdriver.doc - Documentation you are reading now.

#### USAGE

Select OPAL.svdriver as the display driver from SuperViewPrefs and/or from the superview compatible software you are using.

While conversion for OpalVision is taking place, the following information will be displayed:

OPAL.svdriver version number,

and the image size (which can be larger than the screen size).

#### REQUIREMENTS

- 1) An OpalVision card :-)
- 2) opal.library must be in LIBS: otherwise the driver will fail with an internal error.
- 3) A minimum of Kickstart 2.0.

#### CONTROL PAD

ControlPad-Name:

"ENV:SuperView-Library/OPAL.controlpad"

ControlPad-Commands:

OVERSCAN=ON|OFF

Force Opal to display in either overscan or non-overscan. Not specifying an overscan setting informs OPAL.svdriver to intelligently select an overscan screen mode for you.

CENTER\_IMAGE

Force the displayed image to be taken from the centre section of the image buffer, i.e. If the image is 640 x 512 and the display mode is LoRes, Non-interlace and CENTER\_IMAGE is specified then the top left of the visible display will be 160, 156 offset into the original image.

If not present, the image will be displayed from the top left of the image buffer.

#### NOTES

Opal images cannot be scrolled yet (until superview supports scrolling images larger than the drivers screen size), so if the picture is larger than the display area, the displayed image will be cropped.

#### HISTORY

V0.0 First the earth cooled.

V1.0 Initial release.

V2.2 Supports new features of superview.library version 11.6

- Selectable screen modes.
- Control pads for configuration.

V2.3 - Fixed byte aligning of 24 bit images.

- Drastically reduced memory requirements and increased speed by removing the conversion from RGB triplets through RGB planes.

#### COMPLIER

OPAL.svdriver was written using CED V3.5 and compiled with SAS/C 6.55 on an Amiga 2000/030 and Amiga 4000/040. Enforcer was used to detect and correct programming errors.

#### BUGS

Should you find any bugs, please report them so that they can be fixed. Likewise any suggestions for improvement of the driver should be forwarded so that they can be addressed.

#### THANKYOU

Very many thanks to Andreas, the author of Superview.library for implementing some of our suggestions and also for explaining some of the

---

internals of superview.library. Without superview this driver would be a boat anchor. (It would probably float :-)

#### AUTHORS

You can contact the authors via:

Email:

Paul Huxham

paulh@Perth.DIALix.oz.au

Steve Quartly

steveq@sndcrft.DIALix.oz.au

or

P.O. Box 875

Morley,

Perth,

Western Australia 6943

## 1.73 Retina.svdriver

© 1995-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 2.9

Release Date : 01.08.1996

#### Requirements

~~~~~

- OS V2.04+ (V37)+ and its libraries
- superviewsupport.library V6+
- retina.library V7+
- retinaemu.library V1+
- AMIGA with Retina Z2/Z3 Graphics Card and Software

#### Description

~~~~~

Retina.svdriver is an external SV\_GfxBuffer display module for superview.library, which allows object-oriented displaying on Amigas with the 16/24 bit Retina Workbench Emulation System.

The related Retina-libraries are needed.

This Driver supports the following :

| Source   | Depth | Special Modes | Dimensions          | Dest. Depth |
|----------|-------|---------------|---------------------|-------------|
| ONEPLANE | 8/24  | HAM6, HAM8    | [ Retina-Dependent] | 16 / 24     |
| BITPLANE | 1..8  | HAM6, HAM8    | [ Retina-Dependent] | 16 / 24     |

HAM6/8 will be converted to 24 bit, which then will either be displayed as 16 or 24 bit.

#### USAGE

~~~~~

RetinaEmu:

-----

For SuperView's "svdRetina-Screen" screen entry, select an AutoScroll-Area of "800x600" and "No Refresh" with "Retina Standard" (Z2) or "Retina Blitter" (Z3) output.

RetinaScreenMode:

-----

The Monitor specifications need to be 100% correct.

What using instead?

-----

Retina.svdriver always displays graphics in 16/24 bit - so when running on AGA systems, AGA.svdriver would be the better solution for upto 256 Colors. On ECS systems this may only apply to 16/32 Color graphics as long as you don't have OS 3.1 and a RetinaEmu which supports 256 Color Screens for this configuration (should be available). Otherwise with ECS.svdriver you would still have to specify "8BITOPERATOR" for more than 16/32 Colors, which then would be as slow or even slower.

Credits

~~~~~

Thanks to Florian Zeiler (Irseesoft) for supplying the Gfx Card.

The Retina Emulation Software is of course copyrighted by MS MacroSystem Computer GmbH Germany, which is hereby expressively respected in all points.

ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/Retina.controlpad"

ControlPad-Commands : - EMUSCREENDEPTH=<16|24>  
; beginning depth for opening Retina Screens  
; Default is 24 bit (if opening fails, it is  
; also tried to open a 16 Bit Screen, then)

History

~~~~~

V2.9 (01.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using new startup-code now (oops, forgot that before)
- new GST
- using utility.library
- smaller now
- and more

V2.8 (29.10.1995) :

- recompiled with SAS/C V6.56

V2.7 (9.10.1995) :

- fixed version counting for info structure

V2.6 (1.10.1995) :

- COMPLETE, MAJOR REWORK.
-

- fixed several bugs; did some thousand workarounds :-)
- removed several controlpad entries :-)
- fixed width, height and bpl parameters for cases, where screensize and buffersize do differ (larger or smaller than expected/requested/desired)
- fixed screenmode generation for BestModeID generation with graphics, where width<height. Before, always NTSC:LowRes was used (returned by OS function), which caused crashes. Side effect sometimes are better x/y aspect results.
- changed way of autoscroll activation (as for Retina.svdriver it's always active, but RetinaEmu needs to be informed about)
- now always allows to select and specify ScreenModes, as long as compliant to the needs of the Retina (when being passed, they'll be checked. Valid ViewModes also are available about the ViewMode-Requester when using the list supplied by Retina.svdriver, e.g. fromout SuperView)
- requests support-lib V6+ now

```
*****
* Revision history entries before V2.6 have been deleted for space reasons *
*****
```

## 1.74 MERLIN.svdriver

© 1994-1996 by Thomas Eigentler,

© 1996 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 2.4

Release Date : 19.09.1996

### Description

~~~~~

MERLIN.svdriver is an external SV\_GfxBuffer display module for superview.library, which allows object-oriented displaying on Amigas with the 16/24 bit Merlin Workbench Emulation System.

The related Merlin-libraries are needed.

### Original Author

~~~~~

MERLIN.svdriver has been included into the SuperView-Library distribution with friendly permission by the original author, who also supplied the source code to continue development.

Thomas' does no longer continue development of MERLIN.svdriver, which I took over from him, but for any other issues you may Thomas Eigentler reach him via one of his email addresses:

FidoNet: Thomas Eigentler 2:246/1511.0

UseNet: Thomas\_Eigentler@damage.tynet.sub.org

thomas.eigentler@student.uni-tuebingen.de

### Credits

~~~~~

The Merlin Emulation Software is of course copyrighted by its authors (ProDev), which is hereby expressively respected in all points.

Thanks go to Mika Lundell and Thomas Eigentler for reporting bugs and doing beta testing.

#### Notes

~~~~~

It has been reported, that MERLIN.svdriver crashes, when it is tried to open the 040 version of hrgblitter.library on a 060.  
Not a bug of superview.library (-> Thomas Eigentler).

#### History

~~~~~

Here's a short overview of the previous releases of the Driver (concatenated and translated by me, not Thomas - so don't rely on it ;-)

V2.4 (19.09.1996) :

- \*tried\* to fix some well known bugs (8 Bit chunky graphics distorted, with every second line blank) with the help of Mika Lundell (thanks), but it seems, as if the Merlin software itself is buggy :-((

V2.3 (22.08.1996) :

- taken over development from Thomas Eigentler
- now uses/requires semaphore system of superview.library V14+
- using new startup-code now (oops, forgot that before)
- new GST
- using utility.library
- smaller now
- and more

```
*****
* Revison history entries before V2.3 have been deleted for space reasons *
*****
```

## 1.75 XOR.svoperator

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3  
Release Date : 05.08.1996

#### Description

~~~~~

XOR.svoperator is an external SV\_GfxBuffer modification module for superview.library.

XOR.svoperator modifies a graphics in a way, that the color register numbers of the supplied picture's pixels are XOR'ed with a given value, which may be specified via ControlPads (see below).

## ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/XOR.controlpad"

ControlPad-Commands : - XORVALUE=<0..255>  
                               ; 0 and 255 will not be the best decision ;-)  
                               ; Use 15 or something like this.

## History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

\*\*\*\*\*  
 \* Revison history entries before V3.1 have been deleted for space reasons \*  
 \*\*\*\*\*

## 1.76 24BitToHAM.svoperator

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.4

Release Date : 11.08.1996

## Description

~~~~~

24BitToHAM.svoperator is an external SV\_GfxBuffer modification module  
 for superview.library.

24BitToHAM.svoperator dithers 24 bit RGB graphics to HAM6/HAM8,  
 either quick or well.

ControlPad-Name : "ENV:SuperView-Library/24BitToHAM.controlpad"

ControlPad-Commands : - DITHERMODE=< HAM6\_QUICK|HAM6\_WELL  
   |HAM8\_QUICK|HAM8\_WELL>  
                               ; specifies the HAM-Mode to be used and  
                               ; the resulting speed/quality

## History

~~~~~

V3.4 (11.08.1996) :

- fixed progress indicator for "well"-modes also  
(-> Patrik Nydensten)
- highly optimized "ham8\_well" mode (but still is very slow)
  - now completely different from ham6 version (100% rewritten)
- added 68030 version

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56
- now tries to keep the selected, original ViewMode for the 24 bit graphics, if it has HAM capability. Otherwise old behaviour takes place. So if you select "DBLPAL:HighRes", it will no longer become "PAL:HighRes" or "PAL:HighRes Interlace". Was uncomfortable when not using 24BitToHAM separately, but instead hidden behind ECS/AGA.svdriver's 24BITOPERATOR feature.

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

```
*****
* Revision history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.77 Crop.svoperator

© 1995-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3  
Release Date : 05.08.1996

Description

~~~~~

Crop.svoperator is an external SV\_GfxBuffer modification module for superview.library.

Crop.svoperator may extract shapes of any size from any 2..256 Color or 24 bit graphics.

Values for "LeftEdge", "TopEdge", "Width" and "Height", describing the crop box have to be supplied via ControlPad settings.

ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/Crop.controlpad"  
ControlPad-Commands : - CROP\_LEFTEDGE=<Value>  
; crop from x position  
; (will be adjusted, if >= source width)



- CROP\_TOPEDGE=<Value>  
; crop from y position  
; (will be adjusted, if >= source height)
- CROP\_WIDTH=<Value>  
; crop how many x pixels from leftedge  
; (will be adjusted, if too large)
- CROP\_HEIGHT=<Value>  
; crop how many y pixels from topedge  
; (will be adjusted, if too large)

#### History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

\*\*\*\*\*  
\* Revison history entries before V3.1 have been deleted for space reasons \*  
\*\*\*\*\*

## 1.78 Dither24Bit.svoperator

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.5

Release Date : 30.08.1996

#### Description

~~~~~

Dither24Bit.svoperator is an external SV\_GfxBuffer modification module for superview.library.

Dither24Bit.svoperator dithers 24 bit RGB graphics to 256 Colors by default. Possible is also dithering to less colors (2..128), if specified via ControlPad-Settings.

This module applies Heckbert's median cut and dithers using Floyd-Steinberg.

#### ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/Dither24Bit.controlpad"

```
ControlPad-Commands : - COLORDEPTH=<1..8>
                        ; specifies the colordepth of the dithering
                        ; output (1->2 Colors .. 8->256 Colors)
- DITHERMODE=<BESTPEN|DITHER_FLOYD-STEINBERG
                |ORDERED|BURKES>
                        ; whether to just select the best pen or
                        ; do Floyd-Steinberg pixel error adjustment
```

#### History

~~~~~

V3.5 (30.08.1996) :

- added new dithermode: BURKES

V3.4 (28.08.1996) :

- used some `__inline`s instead of `#defines`
- fixed `bestpen()` routine
- added BESTPEN mode (no Floyd-Steinberg dithering)
- (hopefully) fixed bug in F/S error spreading code
- completely replaced F/S dithering by better routine (more reliable, better results, faster)
- added new dither mode: ORDERED

V3.3 (05.08.1996) :

- now uses/requires semaphore system of `superview.library` V14+
- using `utility.library` and new GST
- was linked for `utility.library`, although not compiled for
- using new startup-code now
- no longer allocs large buffers using `MEMF_PUBLIC` (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (09.10.1995) :

- upgraded to V2/V3 specs
- now uses black&white for 2 color dithering mode (-> Florian Zeiler)

```
*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.79 HilbertDither256.svoperator

© 1994-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3

Release Date : 05.08.1996

## Description

~~~~~

HilbertDither256.svoperator is an external SV\_GfxBuffer modification module for superview.library.

HilbertDither256.svoperator modifies any supplied source SV\_GfxBuffer in a way, that all supplied graphics with upto 256 Colors are dithered to Black & White graphics (2 Colors), as e.g. needed for desktop publishing or output on matrix printers.

It uses the fractal Hilbert curve for getting best results in eliminating the resulting errors.

As a side effect, the resulting picture will always have a width and height, which is divideable by 16 (graphics will be adjusted this way).

## ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/HilbertDither256.controlpad"  
 ControlPad-Commands : - BACKGROUND=<BLACK|WHITE>  
                               ; defines, which of the two colors will act  
                               ; as background color. Useful e.g. for printing.

## History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

```
*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.80 AnyTo24Bit.svoperator

© 1995-96 by Andreas R. Kleinert.

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3  
 Release Date : 05.08.1996

## Description

~~~~~

AnyTo24Bit.svoperator is an external SV\_GfxBuffer modification module

for superview.library.

AnyTo24Bit.svoperator converts any input graphics into TrueColor  
RGB graphics with 24 bit.  
Input may be any (upto) 256 Color graphics or HAM6/HAM8 data.

History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs
- HAM8 -> 24 bit conversion had not been fixed in 1.6  
Now it has.  
(-> Florian Zeiler, Jürgen Schäfer)

\*\*\*\*\*  
\* Revison history entries before V3.1 have been deleted for space reasons \*  
\*\*\*\*\*

## 1.81 ExtractGrayScales

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.5  
Release Date : 05.08.1996

Description

~~~~~

ExtractGrayScales.svoperator is an external SV\_GfxBuffer modification module  
for superview.library.

ExtractGrayScales.svoperator modifies any supplied source SV\_GfxBuffer  
in a way, that its colormap will be changed to reflect gray-scales  
(works with (upto) 256 Colors and 24 bit files).  
Output is done in input colordepth or 256 Colors by default (fastest).

This is a sample SVOperator for simple demonstration how to write one.  
It's simple, but effective.

ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/ExtractGrayScales.controlpad"  
ControlPad-Commands : - COLORDEPTH=<1..8>

```

; specifies the colordepth of the grayscale
; output (1->2 Colors .. 8->256 Colors)
- QUICK
; uses >>2, >>1, >>3 (*0.25, *0.5, *0.125)
; instead of *0.3, *0.59, *0.11

```

## History

~~~~~

V3.5 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.4 (21.12.1995) :

- added REV. 2 of new, rock-solid startup code

V3.3 (16.12.1995) :

- added new, rock-solid startup code
- no longer causes crashes
- now using new GST

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

```

*****
* Revision history entries before V3.1 have been deleted for space reasons *
*****

```

## 1.82 ExtractRed

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3  
Release Date : 05.08.1996

### Description

~~~~~

ExtractRed.svoperator is an external SV\_GfxBuffer modification module for superview.library.

ExtractRed.svoperator modifies any supplied source SV\_GfxBuffer in a way, that only the RED values of a picture will be extracted for creation of a new (upto) 256 Color graphics (works with (upto) 256 Colors and 24 bit files).

## History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

\*\*\*\*\*  
\* Revison history entries before V3.1 have been deleted for space reasons \*  
\*\*\*\*\*

## 1.83 ExtractGreen

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3

Release Date : 05.08.1996

## Description

~~~~~

ExtractGreen.svoperator is an external SV\_GfxBuffer modification module for superview.library.

ExtractGreen.svoperator modifies any supplied source SV\_GfxBuffer in a way, that only the GREEN values of a picture will be extracted for creation of a new (upto) 256 Color graphics (works with (upto) 256 Colors and 24 bit files).

## History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

```
*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.84 ExtractBlue

© 1994-96 by Andreas R. Kleinert.

FREEMWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3  
Release Date : 05.08.1996

Description  
~~~~~

ExtractBlue.svoperator is an external SV\_GfxBuffer modification module for superview.library.

ExtractBlue.svoperator modifies any supplied source SV\_GfxBuffer in a way, that only the BLUE values of a picture will be extracted for creation of a new (upto) 256 Color graphics (works with (upto) 256 Colors and 24 bit files).

History  
~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

```
*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.85 TopToBottom

© 1994-96 by Andreas R. Kleinert.

FREEMWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3  
Release Date : 05.08.1996

---

## Description

~~~~~

TopToBottom.svoperator is an external SV\_GfxBuffer modification module for superview.library.

TopToBottom.svoperator modifies any supplied source SV\_GfxBuffer in a way, that it will be swapped from bottom to top (works with (upto) 256 Colors and 24 bit files).

## History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

```
*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.86 LeftToRight

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3  
Release Date : 05.08.1996

## Description

~~~~~

LeftToRight.svoperator is an external SV\_GfxBuffer modification module for superview.library.

LeftToRight.svoperator modifies any supplied source SV\_GfxBuffer in a way, that it will be swapped from left to right (mirrored) (works with (upto) 256 Colors and 24 bit files).

## History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)



V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

```
*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.87 Rotate

© 1995-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3

Release Date : 05.08.1996

Description

~~~~~

Rotate.svoperator is an external SV\_GfxBuffer modification module for superview.library.

Rotate.svoperator rotates any (upto) 256 Color or 24 bit graphics by (default) 90 degrees (reverse clockwise).

The default behaviour may be overwritten via controlpad settings, which also allow 180 and 270 degrees (no extra memory needed: uses different algorithm).

ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/Rotate.controlpad"

ControlPad-Commands : - DEGREES=<90|180|270>

; rotate by how many degrees (reverse clockwise) ?

History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

```
*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.88 RotateFree

© 1995-96 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.5

Release Date : 05.08.1996

Description

~~~~~

RotateFree.svoperator is an external SV\_GfxBuffer modification module for superview.library.

Rotate.svoperator rotates any (upto) 256 Color or 24 bit graphics by 0..360 (default is 90) degrees (reverse clockwise) about any given point (default is the middle, which is width/2, height/2). The default behaviour may be overwritten via controlpad settings.

ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/RotateFree.controlpad"

```
ControlPad-Commands : - ROTATE_ANGLE=<0..360>
                        ; rotate by how many degrees (reverse clockwise) ?
- METHOD=<MIDDLE|GIVENPOINT>
                        ; default is rotation about the middle
- X_COORD=<Value>
                        ; if METHOD=GIVENPOINT we need (X/Y)
                        ; Default is (0/0)
- Y_COORD=<Value>
                        ; if METHOD=GIVENPOINT we need (X/Y)
                        ; Default is (0/0)
- KEEPSIZE
                        ; if this keyword is set, it is NOT tried to
                        ; readjust the image size and to center the
                        ; image, so that parts, which would be out of
                        ; range won't be cut off (which works best
                        ; with (X/Y) pairs in the left, upper quarter)
```

History

~~~~~

V3.5 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.4 (13.1.1996) :

- mathtrans.library was not opened by the code, but AUTOOPENLIB instead -> did crash ALWAYS

- it seems as if the opening (can't say, whether the libraries themselves) of transient math libraries (like "mathtrans.library" together with "mathffp.library" or "mathieeedoubtrans.library" together with "mathieeee.library") did cause all the concerned SVObjects and/or SVOperators to crash on certain systems like 68060, some 68000 and some 68040 (on my system only, when running SnoopDOS at the same time to catch library openings).

No problems when only using "mathffp.library" or using link libraries for both, non-transient and transient functions.  
No more crashes.

We now do that.

- added correct sscanf() routine (.lib)

V3.3 (21.12.1995) :

- added REV. 2 of new, rock-solid startup code

V3.2 (16.12.1995) :

- added new, rock-solid startup code
- replaced IEEE with FFP support (due to crashes)
- no longer causes crashes
- now using new GST

V3.1 (29.10.1995) :

- first version

## 1.89 Scale50

© 1995-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3  
Release Date : 05.08.1996

Description

~~~~~

Scale50.svoperator is an external SV\_GfxBuffer modification module for superview.library.

Scale50.svoperator scales any (upto) 256 Color or 24 bit graphics to their half size by default.

The default behaviour may be overwritten via controlpad settings, which also allows to double the size instead.

ControlPad-Switches

~~~~~

ControlPad-Name : "ENV:SuperView-Library/Scale50.controlpad"  
ControlPad-Commands : - METHOD=<HALF|DOUBLE>  
; scale to which size ?

History

~~~~~

V3.3 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

\*\*\*\*\*  
 \* Revision history entries before V3.1 have been deleted for space reasons \*  
 \*\*\*\*\*

## 1.90 CallPNM

© 1995-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.2

Release Date : 05.08.1996

Description

~~~~~

CallPNM.svoperator is an external SV\_GfxBuffer modification module for superview.library.

This is not an operator in the common sense.

Instead it is just an Interface to allow easy access to any external operators from the well known PBM package (distributed as NetPBM).

When using this operator, the given GfxBuffer will be saved as a 24 bit PNM file (8 Bit input as well), after that the specified PBM program module will be externally called and its output will be written into another temporary file.

This file then will be parsed through superview.library (perhaps it will be in PNM format, but one never know) and the buffer will be returned as the result of the "CallPNM" operation.

Please note, that the whole internal construction of the operator is a little bit more complicated than usual, so that any error handling will perhaps not always result in very clear statements (can't parse the output of the PBM programs).

BTW, any temporary files will of course be placed into "VMEM:", from where they will be deleted later.

ControlPad-Switches

```

~~~~~
ControlPad-Name      : "ENV:SuperView-Library/CallPNM24.controlpad"
ControlPad-Commands : - PNMCOMMAND=<PBM command path and name>
                      ; PNM command to be called
                      ; e.g. PNMCOMMAND=Work:NetPBM/pnm/pnmscale
- PNMOPTIONS=<Options>
                      ; Options for PNM command to be called
                      ; e.g. PNMOPTIONS=-xscale 2.0 -yscale 2.0

```

## History

```
~~~~~
```

V3.2 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- using new startup-code now
  - (oops, forgot that before - other notes were wrong)

V3.1 (07.06.1996) :

- there was a little bit messed up concerning version information
- NOW the module has been: - recompiled with SAS/C V6.56
  - upgraded to V2/V3 specs
- added new, rock-solid startup code
- PNM command now called in double quotes

```

*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****

```

## 1.91 OptimizePalette

© 1995-96 by Andreas R. Kleinert.

FREWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.4  
Release Date : 05.08.1996

### Description

```
~~~~~
```

OptimizePalette.svoperator is an external SV\_GfxBuffer modification module for superview.library.

OptimizePalette.svoperator filters all unused colors out of a palette of a given graphics with upto 256 Colors  
(EHB, HAM6/8 and 24 bit graphics will be rejected).

A new palette will be created - which also misses any duplicate color table entries - to which then the given graphics is remapped.

This Operator may have three effects (plus combinations):

- a) none, if the palette already was or has been optimized
- b) only some black colors at the end of the color map, if

- removing of unused colors did not suffice to fall back to the next lower depth boundary  
( $\leq 256$  to  $\leq 1, 2, 4, 8, 16, 32, 64$  or  $128$ )
- c) a reduced color depth to one of the next lower depth boundaries (as described under b) if there've been a lot of actually unused colors
- d) combinations of b) and c)

#### History

~~~~~

V3.4 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.3 (8.1.1996) :

- added REV. 2 of new, rock-solid startup code
- now using new GST
- the svoperator shifted the image left by one pixel, adding garbage to the right border, when color reduction was performed.  
(-> Henrik Tikanvaara)

V3.2 (29.10.1995) :

- recompiled with SAS/C V6.56

V3.1 (08.10.1995) :

- upgraded to V2/V3 specs

```
*****
* Revison history entries before V3.1 have been deleted for space reasons *
*****
```

## 1.92 PaletteDither.svoperator

© 1996 by Andreas R. Kleinert.

(Also see notes under "Credits".)

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 3.3

Release Date : 28.08.1996

#### Description

~~~~~

PaletteDither.svoperator is an external SV\_GfxBuffer modification module for superview.library.

PaletteDither.svoperator dithers 24 bit RGB graphics 2..256 color graphics by using a selectable number of colors from a specified fixed palette for that (taken from a given graphics file) and calculating the (optional) rest by itself.

This is useful for creation of ANIMations, where the palette for all single frames have to be (mostly) identical or for games and other applications where the palette of used graphics have to harmonize together.

ControlPad-Switches

~~~~~

```
ControlPad-Name      : "ENV:SuperView-Library/PaletteDither.controlpad"
ControlPad-Commands : - COLORDEPTH=<1..8>
                        ; specifies the colordepth of the final,
                        ; dithered graphics (1..8 for 2..256 colors)
- DITHERMODE=<BESTPEN|DITHER_FLOYD-STEINBERG>
                        ; whether to just select the best pen or
                        ; do Floyd-Steinberg pixel error adjustment
- PALETTEDEPTH=<1..8>
                        ; number of colors (depth) to be taken from the
                        ; fixed palette - the possible rest will be
                        ; generated (if specifying more palette colors
                        ; than available, all available will be taken)
- PALETTEFILE=<palette graphics filename>
                        ; any 2..256 color graphics file, of which
                        ; superview.library is able to extract a
                        ; palette from (for example an IFF-ILBM file,
                        ; but including a BMHD and - maybe empty - BODY).
                        ; Allows to take one graphics as sample for
                        ; the others (concerning the palette).
```

History

~~~~~

V3.3 (28.08.1996) :

- used some \_\_inline instead of #defines
- fixed bestpen() routine
- added BESTPEN mode (no Floyd-Steinberg dithering)
- completely replaced F/S dithering by better routine  
(more reliable, better results, faster)

V3.2 (05.08.1996) :

- now uses/requires semaphore system of superview.library V14+
- using utility.library and new GST
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V3.1 (27.01.1996) :

- first version  
(-> Michael Flad)

## 1.93 Requirements for the SuperView-Library Package

Generally, you need at least an 68000 Amiga, running with OS 2.04+. Better performance results require better software/hardware.

More requirements (software, hardware, legal):

```
\textdegree{} For usage of the IFF-based SVOjects, like ILBM, ACBM, PBM, DEEP
and YUVN, you should take care to have iffparse.library V37+
on your bootdisk or system partition.
\textdegree{} For DataType support with OS 3.x+ you need datatypes.library V39 ←
+
to be present. If it is not present, datatype-support is disabled,
but you might not need it, anyway.
\textdegree{} For being able to use the 24 bit datatype extension for ←
CyberGraphX
- via CyberDataType.svobject - you need picture.datatype V43
to be installed together with a registered CyberGraphX system.
If you don't have CyberGraphX, but picture.datatype V43
CyberDataType.svobject will not work, but the datatype V43 may
badly influence the standard picture.datatype support of
superview.library (and several other applications as well).
```

Instead of installing CyberGraphX, you may as well use the freely available cyberncg.library, which comes with picture.datatype V43 - it does replace some parts of the functionality of cybergraphics.library, to make the datatype work on ECS/AGA systems.

```
\textdegree{} Postscript (TM) support via GhostScript.svobject and EPS. ←
svobject
requires Ghostscript being correctly installed and configured.
Without working GhostScript.svobject and a special flag set,
EPS.svobject will not be able to extract any of the Postscript (TM)
data from EPS(F) files, but only the possibly included preview
images from the header (which usually are TIFFs).
\textdegree{} AmiFIG (fig) support requires both, Ghostscript and AmiFIG. ←
svobject,
being correctly installed and configured.
\textdegree{} GPlot (CGM) support requires both, Ghostscript and GPlot. ←
svobject,
being correctly installed and configured.
\textdegree{} GIF and LZW-TIFF support requires a general (non-existent patent
because of countries' law) or personal (if protected by patent
claims in country) license for LZW.
See Installation section for how to apply a given or non-existing
patent situation to this software.
The ability of this software to handle LZW is completely in the
hand of the user - so he is responsible for any licensing to be done.
```

Additionally there are some more requirements, which do not necessarily depend on the OS or the hardware:

```
\textdegree{} Some SVOjects, SVDivers or SVOperators additionally require
different hardware/software configurations, but usually this is
stated within their own documentation.
```

Turbo versions will be installed by the installer-script automatically, if it makes sense (CPU auto-detection - should even work with 68060).

## 1.94 NotesAndHints



## Notes and Hints

How much Memory does this program eat ?!  
 Problems with displaying 24 bit files  
 Problems with converting 24 bit files

### 1.95 Memory Usage

How much Memory does this program eat ?!  
 ~~~~~  
 Simply enough, there are no fixed limits ...

Superview.library and its attached SVOBJECTS, SVDIVERS and SVOperators request as much memory as they need - which directly depends on the size of the processed graphics - and there's no automatic virtual memory manager or such (I added some virtual memory support, but up to now there aren't any modules making use of it).

But actually no CHIP memory is needed for most of the actions (only some chunky-to-planar conversion operations and some specific SVDIVERS may make use of CHIP memory) - so perhaps nearly anything will be performed in FAST ram, when your configuration offers enough. You should have at least 2 MB Ram at all (1 CHIP, 1 FAST or 2 CHIP), better 4 MB (2 CHIP, 2 FAST) or more.

The large memory usage results out of the buffering-technique, which is used with the SuperView-Library.

This technique increases memory usage by using large (full-sized) buffers, which often do exist twice e.g. for conversion from chunky-pixel to bitplane format or vice versa.

But this increases SPEED !!

There are perhaps no "real" multimedia-systems out there, which have less than 6 - 16 MB and why shouldn't we use this memory (you may BTW use VMM or so, if you don't have so much RAM).

Anyway, here's a check-list for what you can do to make more memory available to superview.library :

- o first of all, check if there are some background-programs running, which you do not really need (especially Commodities). Remove them !
- o call "avail flush" in the Shell or select "flushlibs" in the WB-debug menu (available when started with "LoadWB -debug"). (Superview and Image Engineer do also allow flushing Libraries directly fromout the program.)
- o decrease the "AddBuffers" values of drives, which you do not really need (e.g. DF2, DF3, PC0, ...)
- o last not least : leave/close Workbench, stay in the Shell
- o start your favorite viewer or conversion program (e.g. Superview)

- o again: call "avail flush"  
(this time also all unused SVOobjects and SVDdrivers will be flushed out of the memory, so that we later only will have the needed ones in Ram)

If you tried all of the above, and memory still does not suffice, you should buy some more SIMMs at your local Computer Shop...

## 1.96 Displaying 24 bit graphics

### The Problem

-----

"All of my attempts in displaying my really nice 24 bit graphics with AGA.svobject resulted in really ugly colors. What shall I do ?"

### The Reason

-----

You did not set an available or valid 24 bit SVOperator, which could have been used by the selected SVDriver (e.g. AGA.svdriver, ECS.svdriver) adjusting the number of colors to less than 256 (by dithering, conversion to gray, etc.).

### The solution

-----

- o Within the SVDriver's controlpad settings file (e.g. AGA.controlpad, ECS.controlpad) you may set "24BITOPERATOR=<operatorname>" and specify a specific operator for e.g. dithering 24 bit graphics to 256 colors (Dither24Bit), converting them to grayscales (ExtractGrayScales) or HAM6/8 (24BitToHam), etc.
- o The same trick may take place for ECS.svdriver on systems, which can't display more than 16/32 (64EHB) colors, so that specifying the (optional) "8BITOPERATOR=<operatorname>" would allow automatic conversion to less colors.
- o Besides this, a special case perhaps perhaps is the combination of AGA.svdriver and JPEG.svobject, where it may be good for performance and memory usage to simply use the internal on-the-fly dithering of JPEG.svobject, which then simply does not make 24 bit, but 256 color output anymore, when the color depth controlpad switch has been set as: "COLORDEPTH=8" within JPEG.controlpad.

These default settings intuitively can be changed via SVPrefs, or possibly via your application's GUI as well.

- o An other possibly way is, to always do the dithering only when needed - by simply invoking the needed SVOperator fromout your application's GUI.

## 1.97 Converting 24 bit graphics

---

### The Problem

-----

You have problems converting from one 24 bit file format into other 24 bit file formats ?

### The Reason

-----

Not necessarily all SVOjects, which support reading of 24 bit files do also support writing of 24 bit files.

SVOjects, which support writing 24 bit data yet are for example:

|      |           |       |
|------|-----------|-------|
| ILBM | JPEG      | Targa |
| BMP  | SunRaster | YUVN  |
| PCX  | FBM       | PNM   |
| SVG  | SGI       | DEEP  |
| TIFF | QRT       |       |

### The solution

-----

A temporary solution might be to use other 24 bit programs for conversion, if you don't want to use one of the file formats, which are already supported.

## 1.98 SVPrefs

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved. Only to be distributed with SuperView-Library.

Version : 13.3  
Release Date : 30.08.1996

### Description

~~~~~

SVPrefs is the Preferences Program for superview.library.

Any specific ControlPad settings for superview.library, superviewsupport.library and all the SVOjects, SVDdrivers and SVOoperators may be modified from out here in a comfortable GUI-based way - as long as these ControlPad requirements are documented in form of a CPInfo-file placed in ENVARC:SuperView-Library (as usually).

Additionally, adding and removing of single SVOjects, SVDdrivers or SVOoperators is possible from here, if no other program is accessing superview.library and the related modules at the same time (to keep the internal lists valid).

### Inclusion and Exclusion: Explanation

-----

As one of the newest features of superview.library, you may also decide here, which of the installed SVOjects, SVDdrivers and/or SVOoperators should not be loaded into memory at startup time. This works as follows:

- excluded modules will be written to a specific configuration file at ENVARC:superview-Library and ENV:superview-Library (svobjects.exclude, svdrivers.exclude, svoperators.exclude)
- excluded modules will not be opened when the library starts its initialization phase, which means that these modules will never be loaded. May speed up opening and save memory. On the other side, these modules are not available as long as you don't re-add them to the system, either only temporarily or also by re-including. So you only should apply this to modules you seldomly or never use, but don't want to de-install (everytime) nevertheless.
- any of these modules may temporarily be loaded via the "Add" function of SVPrefs as well as these (or others) may again temporarily be removed
- SVPrefs can apply (re)inclusion and exclusion to modules currently held in memory (to re-include an currently excluded module you at first have to re-load it via "Add")

#### Inclusion and Exclusion: Overview

|                     |                 |
|---------------------|-----------------|
| INCLUDE temporarily | ADD             |
| permanently         | ADD, INCLUDE    |
| EXCLUDE temporarily | REMOVE          |
| permanently         | EXCLUDE, REMOVE |

If you don't need that feature: simply don't use it...

#### History

~~~~~

SVPrefs' version depends on the version of  
superview.library it first has been included with.

#### V15.2 (30.08.1996) :

- processing of filerequester-based .cpinfo tags now more sensitive for file name and path conventions, also better remembers former entries, better handles default settings and also performs safe SVOobject, SVDriver, SVOoperator selection from LIBS:#?
- list of SVOobjects, SVDdrivers and SVOoperators now presented alphabetically sorted
- fixed bug in SVOoperator list initialization: did not show the first, but the last entry at startup (luckily not an EHit)

#### V13.3 (28.06.1996) :

- reworked GUI again: smarter, smaller, more handy (-> Simon Edwards)
- since Add/Del and Include/Exclude are now named +/- and I/E, there's no practical use for gadget underscores any longer. Removed.
- removed priority setting facility for SVOobjects. Was not of practical need for the typical user, anyway

- (more useful maybe for debugging purposes, but I never needed it)
- always free'd SVPList as SVOList
- (well, thanks to FreeVec(), that there did not happen anything)
- changed compiler options, no more printf(), etc.
- Saves about 6K. Wow ;-)
- no longer reacts on CLI parameter "?" (since does not make sense)
- did not do any error output (should have gone to CLI stdout).
- Now opening requester, if possible.

V13.2 (06.06.1996) :

-----

- adjusted window size/font-handling to that one of SuperView
- (-> Patrik Nydensten)
- there was still version information referring to 12.x

```
*****
* Revision history entries before V13.2 have been deleted for space reasons *
*****
```

## 1.99 SuperViewSupport-Library

superviewsupport.library

© 1994-96 by Andreas R. Kleinert.

FREEWARE. All rights reserved.

Version : 7.2  
Release Date : 10.09.1996

### Description

~~~~~

superviewsupport.library contains functions, which are heavily used by the superview.library and its SVOjects, SVDdrivers and SVOperators.

superviewsupport.library helps saving disk space by just holding this functions for usage by the other libraries, also some superview.library debugging functions are included.

### ControlPad-Switches

~~~~~

```
ControlPad-Name      : "ENV:superview-library/superviewsupport.controlpad"
ControlPad-Commands : - C2P=<OS|SV>
                      ; determines, whether chunky to planar conversion
                      ; is managed via the appropriate OS functions
                      ; or via internal ones.
                      ; Due to various problems with the OS functions
                      ; "SV" is now default.
                      ; Be careful when using "OS" with GfxCards,
                      ; which carelessly patched graphics.library.
```

### History

~~~~~

V7.2 (10.09.1996) :

- again revised startup-code
- caching SysBase now (SAS/C's \_\_USE\_SYSBASE)
- removed any remaining Assembler fragments  
(there's now only an ULONG remaining,  
which contains 4 Byte "pure" 68k-Code  
for RawDoFmt)
- a little bit smaller and faster now
- Calling ScreenToBitPlane8() now checks for BMF\_STANDARD  
and does an internal ScreenToOnePlane8() plus OnePlaneToBitPlane8()  
for any graphic card screens, which don't offer a standard  
bitmap image. This uses additionally memory (for this case only),  
but prevents us from any crashes because of chunky-only RTG screens,  
since ScreenToOnePlane8() is protected against that

V7.1 (05.08.1996) :

- forgot to mention V6.14 (what did I change ?! ;-)
- slightly changed library-deinitialization (order and behaviour)
- added function for conformeous SV\_GfxBuffer (plus .svgfx\_Buffer)  
allocation (version-dependant)
- no longer allocs large buffers using MEMF\_PUBLIC (for VMM users)

V6.13 (13.04.1996) :

- fixed grabbing of interlaved screens with ScreenToBitPlane()  
(-> Simon Edwards)

V6.12 (29.02.1996) :

- ScreenToOnePlane() still had bugs (same as early ECS, AGA svdrivers,  
but just for reading instead of writing). Fixed alignment, using  
separate buffer now. Also removed some redundancies and usage  
of UWORDS (now possibly faster on 680x0 machines, x>1).  
Fixes saving directly from Screens.  
(-> Simon Edwards)
- BestModeID() now never returns INVALID\_ID anymore. Never.
- added new snail+email adress to this doc

V6.11 (19.12.1995) :

- now does no longer drop comments and linefeeds when loading/saving  
(-> Steve Quartly)  
See includes and docs for more information.

V6.10 (29.10.1995) :

- recompiled with SAS/C V6.56

V6.9 (29.7.1995) :

- as long as either PAL or NTSC is available, SVSUP\_GetBestModeID()  
will no longer produce IDs using DEFAULT\_MONITOR (mask 0x0000FFFF).  
Should help promoting PAL/NTSC to >31KHz.

\*\*\*\*\*  
\* Revision history entries before V6.9 have been deleted for space reasons \*

---

\*\*\*\*\*

## 1.100 SuperView in the Press

=====

P R E S S F E E D B A C K (20.8.1996)

=====

Following some excerpts from articles - mainly from the computer press - which directly or indirectly reported about SuperView(/Library) in the past (if german, these have not been translated).

- Amiga Computing (IDG Media, UK)
- Amiga Magazin, Amiga Public Domain Sonderheft (Magna Media, Germany)
- Amiga Plus (ICP, Germany)
- Amiga special (media, Germany)

\*\*\*\*\*

Amiga Computing (IDG Media, UK)

-----

"The SuperViewLibraries [...] allow other programs, that support SuperView to easily load, save, convert and process images with the minimum of ease. This allows programmers to concentrate on perfecting their program without having to worry about supporting every different type of picture format out there."  
[ Issue 12/1995, page 21. Referred to SuperView V5.10 + Library V12.3.  
The CoverDisk contained ImageEngineer V2.1a and licensed V12.3 ]

"SuperView is a set of libraries, that allows Image Engineer to load and save a large number of different file types, [...]"  
"Installation of SuperView is very straightforward using the standard Amiga installer program, [...]"  
[ Issue 7/1996, page 18/20. Referred to SuperView V5.41 + Library V12.9.  
The CoverDisk contained ImageEngineer V3.0 Demo and licensed V12.9 ]

Amiga Magazin (Magna Media, Germany)

-----

"Zum Betrachten von Bildern ist dieses PD-Programm wohl die beste Lösung. [...]"  
[ Issue 12/1994, p. 120. Referred to SuperView V2.1 ]

\*\*\*

"[...] SuperView besticht durch seinen modularen Aufbau. [...]  
SuperView kennt sehr viele Bildformate. Jedes Format wird durch ein SVOobject verwaltet. Das erlaubt eine flexible Erweiterbarkeit des Programms. [...] Daß das Programm [...] alle Features des Amiga-OS bis hin zur Version 3.1 unterstützt [...] ist angesichts des großen Funktionsumfanges klar. [...]  
Fazit: SuperView ist ein sehr leistungsfähiger Bildanzeiger und eine echte Konkurrenz [...]"  
[ Issue 7/1995, p. 134. Referred to SuperView V4.52 on Time #386 ]

\*\*\*

"[...] Das Programm ist komplett modular aufgebaut und somit

erweiterbar. Sogar die Benutzerschnittstelle ist austauschbar. [...]"  
[ Issue 2/1996, p. 50. Referred to SuperView V5.10 on German #561 ]

\*\*\*

"'SuperView' als Bildanzeiger zu beschreiben, wäre nur die halbe Wahrheit. Diese Aufgabe erfüllt das Programm zwar mit Bravour, hält aber noch anderes auf Lager. [...]"  
[ Issue 4/1996, p. 92. Referred to SuperView V5.32 on Time #486 ]

Amiga Plus (ICP, Germany)

-----  
"SuperView benutzt zwar auch ab OS3.0 Datatypes, zeigt aber ab OS2.0 auch bereits IFF-, [...] -Grafiken an. Erreicht wird dies durch eine eigene 'superview.library', die [...] es ermöglicht, Grafiken in diesen Formaten zu speichern"  
[ Issue 7/1994, S. 42. Referred to SuperView V2.1 ]

\*\*\*

"Wir haben für Sie eine Auswahl der besten und wichtigsten Tools und Utilities aus allen Sparten des Computeralldtags zusammengetragen" [...]  
"SuperView [...] Bildanzeiger, der ab OS3 Datatypes zum Erkennen der Grafikformate verwendet. Mit Hilfe der SuperView-Library sind die wichtigsten Formate auch ab OS2 lesbar."  
[ Issue 4/1995, p. 46. Referred to SuperView V2.1 ]

\*\*\*

[ Formatvielfalt, Konvertierung ]  
"[...] Dieses Kunststück verdankt unser Testkandidat unter anderem der exzellenten SuperView.library von Andreas Kleinert [...]"  
[ Issue 2/1996, p. 73. Article about Picture Manager professional (PMPPro) V3.0, with superview.library 12.x ]

\*\*\*

"SuperView [...] ist ein Anzeiger, der durch unzählige externe Module erweiterbar ist und alle erdenklichen Bildformate auf ECS- und AGA-Amigas, sowie auf EGS-, OpalVision, Picasso-, Merlin-, Retina- und CyberGraphX-Grafikkarten darstellt. [...]"  
[ Issue 3/1996, p. 61. Referred to SuperView V4.63, V4.70, V5.10 ]

\*\*\*

"'SuperView' zeichnet sich durch seine Vielseitigkeit aus. [...] Mehr als nur ein kleiner Bonus sind die 31 Operatoren [...]. Selbstverständlich lassen sich die Bilder auch wieder speichern. [...]"  
[ Issue 6/1996, p. 43. Referred to SuperView V5.32 ]

\*\*\*

"[...] Dieses Programm unterstützt eine deutlich höhere Anzahl von Bildformaten (wie PCX, BMP und TIFF) und kann auch Effekte hinzufügen und Bilder zwischen verschiedenen Formaten konvertieren [...]"  
[ Issue 9/1996, p. 58. Referred to SuperView V5.30 ]

Amiga special (media, Germany)

-----  
"SuperView - Bildbetrachter" [...]

---



"Bildanzeiger gibt es viele [...]  
 Doch ein Programm, das all dies beherrscht und dabei noch  
 anwenderfreundlich und kompakt ist, das fehlte bisher.  
 Die Situation hat sich nun geändert [...]"

"Superview.library [...] Programmierer können mit diesen frei  
 vetreibbaren Routinen interessante und komplexe Anwendungen  
 realisieren. [...] Sämtliche Module sind [...] gut dokumentiert,  
 so daß sich jeder Interessierte schnell zurechtfindet [...]"

"Die grafische Benutzeroberfläche [...]"

"Diese [...] ist einleuchtend aufgebaut, so daß man nach kürzester  
 Einarbeitung damit klarkommt." [...]

"Fazit [...] SuperView ist ein tolles Paket zum Anzeigen und  
 Bearbeiten von Grafiken [...]. Anwender haben mit SuperView endlich  
 einen kleinen und leistungsfähigen Bildanzeiger gefunden, der [...] sogar  
 kleine Funktionen zur Bildbearbeitung enthält."

[ Issue 1/1995, p. 62-63. Article about SuperView V4.0 ]

\*\*\*

"Gerade auch die Unterstützung von Grafikkarten geschieht  
 "mit SuperView einfach und effizient." [...]  
 [ Issue 6/1995, p. 77. Article about SuperView V4.0 ]

\*\*\*

"Im Zusammenspiel mit der 'Superview.library' von Andreas Kleinert  
 konnte diese Weiterentwicklung [...] seine Leistungsfähigkeit  
 eindrucksvoll unter Beweis stellen"

[ Issue 12/1995, p. 116. Article about Picture Manager Professional  
 Beta 0.97 ]

\*\*\*

"Superview gehört mit zu dem Besten, was im Amiga-Sektor an  
 Grafikbetrachtungs-Programmen verfügbar ist. Es ist kompatibel  
 mit fast allen gängigen Formaten [...], besitzt [...] eine  
 durchdachte und komfortable Oberfläche. Erfreulicherweise trägt  
 die mehrsprachige umfangreiche Anleitung [...] und das durchdachte  
 Installationsprogramm ebenfalls zum hervorragenden Gesamteindruck  
 von 'Superview' bei."

[ Issue 5/1996, p. 66. Article about SuperView V5.30 on Time #472 ]

Amiga Public Domain Sonderheft (Magna Media, Germany)

"'Superview' ist sowohl eine Bibliothek (Library), die Lade- und  
 Speicherformate [...] zur Verfügung stellt, als auch Treiber zum  
 Anzeigen von Bildern. [...] Das Ganze erinnert an die Datatypes  
 des Amiga-OS 3.0, allerdings auf Grafik spezialisiert und um  
 Operatoren und spezielle Grafiktreiber erweitert. Außerdem kann  
 die Library mit 24-Bit-Bildern umgehen." [...]

"In der Library steckt viel Arbeit." [...]

"Fazit: SuperView besitzt potentiell das Zeug zum universellen  
 Bildanzeigeprogramm und Konvertierutility." [...]

[ Issue 2/1995, p. 43. Article about SuperView V4.52 / V10.2 ]

## 1.101 Books and other written stuff used during development

- [ 1] "Bitmapped Graphics", 2nd Edition, Steve Rimmer, Windcrest/McGraw-Hill,  
© 1993 by Windcrest Books (registered Trademark of TAB Books).  
ISBN 0-8306-4209-9
- [ 2] "Supercharged Bitmapped Graphics", Steve Rimmer, Windcrest/McGraw-Hill,  
© 1992 by Windcrest Books (registered Trademark of TAB Books).  
ISBN 0-8306-3788-5
- [ 3] "Das Handbuch der Grafikformate", Klaus Holtorf,  
© 1994 Franzis-Verlag GmbH, München  
ISBN 3-7723-6392-X
- [ 4] "Amiga Magazin", Issue 2/1992, Markt & Technik Verlag AG
- [ 5] "DOS Extra", Issue 4/1993, DMV-Verlag
- [ 6] "Das Aufsteigerbuch" (C64 -> Amiga), Michael Strauch, Alexander  
Stellmach, © 1987 by DATA BECKER GmbH, Düsseldorf.  
ISBN 3-89011-134-4
- [ 7] "Formats.doc" of the ShowVIC distribution on SaarAG-Disk #616.  
ShowVIC is (C)opyright 1993 by Matt Francis.
- [ 8] "Einführung in die digitale Bildverarbeitung", Wolfgang Abmayr,  
© 1994 B.G. Teubner, Stuttgart.  
ISBN 3-519-06138-4
- [ 9] "Noch mehr Dateiformate", Günter Born,  
© 1995 Addison-Wesley (Deutschland) GmbH.  
ISBN 3-89319-757-5
- [10] The Independent JPEG Group's software package(s) with Source-Code(s)  
and Documentation. Release 4 through 6.
- [11] "The Programmer's PC Sourcebook", Thom Hogan,  
published by Microsoft Press, © 1991 by Thom Hogan.  
ISBN 1-55615-321-X
- [12] "C/C++ Users Journal (tm)" (several issues),  
© Miller Freeman Inc.
- [13] ... and perhaps books/magazines/articles, which I don't remember yet !
- [14] ... as well as texts found on AmiNet, BBS or CD-ROM.
- [15] Newsgroups in Fido (mostly \*.GER) and UseNet (mostly DE.\* and Z-Netz)

...plus a bunch of Amiga-related books, like RKMs and the Guru Book, etc.

## 1.102 Other Program Projects

| Name<br>of Program    | Archive<br>Name (current)                              | Location<br>(maybe) | Current<br>Version |
|-----------------------|--------------------------------------------------------|---------------------|--------------------|
| ak_gen0-lib           | ak_gen0-lib_38_21Dev.LHA<br>+ ak_gen0-lib_38_21Usr.LHA | -                   | V38.21             |
| AKCC                  | AKCC41.LHA                                             | Aminet:util/cli     | V4.1               |
| C Exec Lib Sample     | CExecLib13.LHA                                         | Aminet:dev/c        | V1.3               |
| C Datatype Sample     | C_OS3-DT.LHA                                           | Aminet:dev/c        | V40.2              |
| DeTAR port            | DeTar12.LHA                                            | Aminet:util/arc     | V1.2               |
| DRAFU plus            | DRAFU105.LHA                                           | Aminet:misc/math    | V1.05              |
| fractal (Y. Fisher)   | frac10.LHA                                             | Aminet:gfx/misc     | 1.0                |
| GNUTar port (V1.11.2) | GNUTar13.LHA                                           | Aminet:util/arc     | V1.3               |
| IFF-Arranger          | IFFArr31.LHA                                           | Aminet:util/misc    | V3.1               |
| K8SVX                 | K8SVX_V2.00.LHA                                        | -                   | V2.00              |
| KFracPlus             | KFrac50.LHA                                            | Aminet:gfx/fract    | V5.0               |
| KILBM                 | KILBM_V1.32.LHA                                        | -                   | V1.32              |
| KVOC                  | KVOC_V1.00.LHA                                         | -                   | V1.00              |

|                    |                     |                   |        |
|--------------------|---------------------|-------------------|--------|
| PR                 | PR_V3.02.LHA        | -                 | V3.02  |
| RetinaView         | RetinaView11-11.LHA | Aminet:gfx/board  | V11.11 |
| SIP                | SIP32.LHA           | Aminet:util/moni  | V3.2   |
| SpaceArchivPlus    | SPAPlus_V3.00.LHA   | -                 | V3.00  |
| STPlayer           | STPlayer_V1.26.LHA  | -                 | V1.26  |
| superplay-lib      | splib51U.lha        | Aminet:mus/play   | V5.1   |
|                    | + splib51D.lha      |                   |        |
| SuperView          | SView563.LHA        | Aminet:gfx/show   | V5.63  |
| SVG.datatype       | SVGDT402.LHA        | Aminet:util/dtype | V40.2  |
| akJFIF.datatype    | akJFIF431.LHA       | Aminet:util/dtype | V43.1  |
| TICKer             | TICKer12.LHA        | -                 | V1.2   |
| UnARJ port (V2.41) | UnARJ241.LHA        | Aminet:util/arc   | V2.41  |
| UtahRLE.svobject   | svoUtah41.LHA       | Aminet:gfx/show   | V4.1   |

Some of the old, obsolete projects have not been uploaded to Aminet from my side. Maybe someone else did, but most possibly are not of such a high interest for today's AmigaOS ;-)

## 1.103 Credits

24BitToHAM.svoperator  
-----

This SVOperator bases on code, which has been included with FBM Release 1.0 25-Feb-90 by Michael Mauldin. The original code had been written by Harald C. Koch to convert 24 bit RGB data (FBM format) to HAM6-ILBM files. I modified it to create format-independent 8 Bit chunky pixel buffers, which can be handled by SuperView-Library.

The code used for the "quick" option of the SVOperator (without palette) is not related in any way to code of the FBM package. The ham8-well code originally was derived from fbham.c, but now no longer does show any similarities to it.

Here's the copyright notice as found in "fbham.c" (revision headers cut off / left out) :

```
* fbham.c: FBM Release 1.0 25-Feb-90 Michael Mauldin
*
* Copyright (C) 1989,1990 by C. Harald Koch & Michael Mauldin.
* Permission is granted to use this file in whole or in part for
* any purpose, educational, recreational or commercial, provided
* that this copyright notice is retained unchanged. This software
* is available to all free of charge by anonymous FTP and in the
* UUNET archives.
[...]
```

```
*
* based on ray2.c from DBW_Render, Copyright 1987 David B. Wecker
*
[...]
```

AmiFIG.svobject

-----  
This FIG support module accesses the external program "fig2dev", which for example is available as port of the fig port AmiFIG 1.1 from AmiNet, which is (C) 1996 Andreas Schmidt.

#### C64.svobject

-----

For getting information about the C64 Koala and Doodle formats I took a look into the source code of ComView 1.0 by Paul Grebenc, which can be found as "C64View" on the SaarAG-Disk #523.

I did not include and use the Source Code as such, but I really learnt a much out of it. The algorithms are perhaps nearly the same, but because I do not use file-to-screen decoding my code is perhaps some 100% faster (different structure, many optimizations).

#### EPS.svobject

-----

EPS.svobject uses strongly modified code from

```
/* pnmtops.c - read a portable anymap and produce a PostScript file
**
** Copyright (C) 1989 by Jef Poskanzer.
**
[...]
```

which is allowed by its copyright statements.

#### FBM.svobject

-----

For getting information on the FBM-Format I took a look into the file "fbm.h" and other source-files, which describe this file format and are part of FBM Release 1.0 25-Feb-90 by Michael Mauldin. No source-code from this package - only the "pure information" - has been used for FBM.svobject.

#### GIF.svobject

-----

For the GIF LWZ Decoding Routines I used some code of the FBM Package. The code of the original routines has been strongly modified and enhanced/improved (there are almost no similarities to the original code left now).

Here's the Copyright notice as found in the file "flgifr.c" (revision headers cut off / left out) :

```
* flgifr.c: FBM Release 1.0 25-Feb-90 Michael Mauldin
*
* Modifications to GIFTORLE are Copyright (C) 1989,1990 by Michael
* Mauldin. Permission is granted to use this file in whole or in
* part for any purpose, educational, recreational or commercial,
* provided that this copyright notice is retained unchanged.
```

---

```

* This software is available to all free of charge by anonymous
* FTP and in the UUNET archives.
*
* Derived from 'giftole', written by David Koblas
*
* +-----+
* | Copyright 1989, David Koblas. |
* | You may copy this file in whole or in part as long as you |
* | don't try to make money off it, or pretend that you wrote it. |
* +-----+

```

For the GIF LWZ Encoding Routines I used some code of the FBM Package.  
The code of the original routines has been strongly modified and  
enhanced/improved.

Here are the Copyright notice of these modules as found in the files  
"flgife.c" and "flgifc.c" (revision headers cut off / left out) :

```

* flgife.c: FBM Release 1.0 25-Feb-90 Michael Mauldin
*
* Modifications to GIFENCODER are Copyright (C) 1989,1990 by
* Michael Mauldin. Permission is granted to use this file in whole
* or in part for any purpose, educational, recreational or commercial,
* provided that this copyright notice is retained unchanged.
* This software is available to all free of charge by anonymous
* FTP and in the UUNET archives.
*
* flgifc.c: FBM Release 1.0 25-Feb-90 Michael Mauldin
*
* Modifications to GIFENCODER are Copyright (C) 1989,1990 by
* Michael Mauldin. Permission is granted to use this file in whole
* or in part for any purpose, educational, recreational or commercial,
* provided that this copyright notice is retained unchanged.
* This software is available to all free of charge by anonymous
* FTP and in the UUNET archives.
*
* Based on: compress.c - File compression ala IEEE Computer, June 1984.
*
* Spencer W. Thomas      (decvax!harpo!utah-cs!utah-gr!thomas)
* Jim McKie              (decvax!mcvax!jim)
* Steve Davies           (decvax!vax135!petsd!peora!srd)
* Ken Turkowski          (decvax!decwrl!turtlevax!ken)
* James A. Woods         (decvax!ihnp4!ames!jaw)
* Joe Orost              (decvax!vax135!petsd!joe)
*

```

GhostScript.svobject

-----

GhostScript.svobject accesses an external Ghostscript port,  
like for example Ghostscript 3.53 from AmiNet, which is  
based on Aladdin Ghostscript.  
Aladdin Ghostscript is Copyright (C) 1989, 1995 Aladdin Enterprises.  
All rights reserved.

---

## Limbo.svobject

-----

Limbo.svobject does support the Limbo 4.0 port from Aminet:gfx/conv, which was based on a version developed by Carsten Frigaard, Jess Gade, Thomas Therp Hemmingsen and Torben Sand in 1993/94 on Aalborg University, Denmark.

## JPEG.svobject

-----

This software is based in part on the work of the Independent JPEG Group. Release 6 was used as found on AmiNet, former attempts based on V4.

## PICT.svobject

-----

The PICT loader module is based on code derived from the PBM package, namely the standalone-module "picttoppm.c". It is said it can be used freely, so I decided to do this instead of trying to implement this weird PICT-2 stuff of QuickDraw (is it a vector format or a bitmap format ;-)

The original source has been strongly modified to fulfil the needs of being integrated as a library module for superview.library. To be more independent from the pbm.package (great improvements on code size) I also extracted the standard bdf font from the pbm part (also free).

Here are the Copyright notice of these modules as found in the files "picttoppm.c" and "libpbm5.c" (revision headers cut off / left out) :

```
/*
 * picttoppm.c -- convert a MacIntosh PICT file to PPM format.
 *
 * [...]
 *
 * Copyright 1989,1992,1993 George Phillips
 *
 * Permission to use, copy, modify, and distribute this software and its
 * documentation for any purpose and without fee is hereby granted, provided
 * that the above copyright notice appear in all copies and that both that
 * copyright notice and this permission notice appear in supporting
 * documentation. This software is provided "as is" without express or
 * implied warranty.
 *
 * George Phillips <phillips@cs.ubc.ca>
 * Department of Computer Science
 * University of British Columbia
 *
 * $Id: picttoppm.c,v 1.7 1993/10/26 22:40:31 phillips Exp phillips $
 */
```

Only the default font definition has been taken from this one:

```
/* libpbm5.c - pbm utility library part 5
**
```

```
** Font routines.
**
** Support for BDF fonts Copyright 1993 by George Phillips.
**
** Copyright (C) 1991 by Jef Poskanzer.
**
** Permission to use, copy, modify, and distribute this software and its
** documentation for any purpose and without fee is hereby granted, provided
** that the above copyright notice appear in all copies and that both that
** copyright notice and this permission notice appear in supporting
** documentation. This software is provided "as is" without express or
** implied warranty.
*/
```

PNG.svobject

-----

Is based on the png reference library (including libpng and zlib), which allows being used e.g. for freely distributable and commercial programs

libpng:

Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.

Contributing Authors:

Andreas Dilger  
Dave Martindale  
Guy Eric Schalnat  
Paul Schmidt  
Tim Wegner

zlib:

(C) 1995-1996 Jean-loup Gailly and Mark Adler

RotateFree.svoperator

-----

The basic algorithm for rotation about any given point with any possible angle had been described in "C/C++ Users Journal", issue August 1995. It was strongly dependend on TIFF input, did only work with 8 Bit input and was not able to adjust the image buffer size to prevent image parts to be cut off. Used only the basic algorithm as a hint how to do hit.

TIFF.svobject

-----

As with version V3.7 this module makes use of the freely distributable TIFF Library (libtiff), which also is used in conjunction with the NetPBM package.

As I can't all files' copyright notices here, and also version information was difficult to detect, I hereby simply include the version headers of the files "README" and "tiff.h", as well as the copyright notice found in tiff.h:

tiff.h,v 1.40 93/08/26 14:28:21

---

README,v 1.17 93/08/26 14:58:20

```
/*
 * Copyright (c) 1988, 1989, 1990, 1991, 1992 Sam Leffler
 * Copyright (c) 1991, 1992 Silicon Graphics, Inc.
 *
 * Permission to use, copy, modify, distribute, and sell this software and
 * its documentation for any purpose is hereby granted without fee, provided
 * that (i) the above copyright notices and this permission notice appear in
 * all copies of the software and related documentation, and (ii) the names of
 * Sam Leffler and Silicon Graphics may not be used in any advertising or
 * publicity relating to the software without the specific, prior written
 * permission of Sam Leffler and Silicon Graphics.
 *
 * THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY
 * WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
 *
 * IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR
 * ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND,
 * OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
 * WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF
 * LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE
 * OF THIS SOFTWARE.
 */
```

YUVN.svobject and PCD.svobject

-----  
For the 24 bit YUV <-> RGB Conversion Routines I used some code of the  
NetPBM Package.

The code of the original routines has been strongly modified and  
enhanced/improved (there are almost no similarities to the original  
code left now, except the main algorithm).

For PCD also the factors to scale YCbCr to RGB had to be adjusted, since  
the ones of PCD are slightly different to those of JPEG.

Here are the Copyright notices as found in the specific source files :

```
/* ppmtoyuvsplit.c - convert a portable pixmap into 3 raw files:
** - basename.Y : The Luminance chunk at the size of the Image
** - basename.U : The Chrominance chunk U at 1/4
** - basename.V : The Chrominance chunk V at 1/4
** The subsampled U and V values are made by arithmetic mean.
**
** If CCIR601 is defined, the produced YUV triples are scaled again
** to fit into the smaller range of values for this standard.
**
** by A.Beck
** Internet: Andre_Beck@IRS.Inf.TU-Dresden.de
**
** Based on ppmtoyuv.c
**
** Permission to use, copy, modify, and distribute this software and its
** documentation for any purpose and without fee is hereby granted, provided
```



```

** that the above copyright notice appear in all copies and that both that
** copyright notice and this permission notice appear in supporting
** documentation. This software is provided "as is" without express or
** implied warranty.
*/

/* yuvsplittoppm.c - construct a portable pixmap from 3 raw files:
** - basename.Y : The Luminance chunk at the size of the Image
** - basename.U : The Chrominance chunk U at 1/4
** - basename.V : The Chrominance chunk V at 1/4
** The subsampled U and V values are made by arithmetic mean.
**
** If ccir601 is defined, the produced YUV triples have been scaled again
** to fit into the smaller range of values for this standard.
**
** by Marcel Wijkstra <wijkstra@fwi.uva.nl>
**
** Based on ppmtoyuvsplit.c
**
** Permission to use, copy, modify, and distribute this software and its
** documentation for any purpose and without fee is hereby granted, provided
** that the above copyright notice appear in all copies and that both that
** copyright notice and this permission notice appear in supporting
** documentation. This software is provided "as is" without express or
** implied warranty.
*/

```

HilbertDither256.svoperator

-----

The description of the "fractal Hilbert dithering" method has been found in the "mc magazine, issue 6/94, Franzis-Verlag GmbH". The basic techniques (L-System, 16x16 block error approximation) are perhaps the same, but the source code as such has been rewritten completely in almost any of its parts, so that it has not just been "taken out of there" (any limits have been removed; e.g. the original source was a standalone-program, which only worked with TARGA graphics, which had a width and height divideable by 16, but max. 1280x960 or 2560x1920).

## 1.104 ControlPad Fileformat

Many SVObjects, SVDivers and SVOperators have to handle globally set preferences, which may also have to be handled and/or modified by custom programs or SuperViewPrefs.

To prevent multiple different ways of storing these preferences data, I introduced a common standard fileformat for such configuration files: The ControlPad fileformat (please read the notes under "Style Guide" below).

- To allow easy reading and understanding of these files, these are stored in plain ASCII
- To prevent people from learning just another technique of configuring files, these are constructed just like a kind of "tooltype lists"

- a single ControlPad takes a full line of the file and looks like that : <Name>=<Content> or <Keyword>  
Content and Name may contain any ASCII characters except "=", because the first "=", which is found, is used to separate the Name from the Content in the line. A Keyword is anything, which does not contain a "=" somewhere, but is not a comment.
- So all other special characters are possible (although you should not make too much use of them for ergonomical reasons).
- it is supported to write as much comments to the files, as wished. A comment line has to begin with "/" or ";" in its first column. A totally blank line fits to the same category as "/" or ";" lines.
- any single line should not exceed 255 characters

#### Additional Notes

~~~~~

- \* Note, that comments are totally ignored by the library functions, when reading ControlPad files, and writing of comments is not supported.

So do not add too much comments to your files, but on the other hand, explain all of the Switches (and their default values) inside the doc-files of your program to prevent a lack of information !

- \* The SVSUP\_FindControlPad() function is case-sensitive, so if nothing else is specified, the ControlPad entries are always also case-sensitive. Case-insensitive functions already have been added (SVSUP\_FindControlPadNoCase()) but are not used widely yet.

By using GUI-based functions for ControlPad modifications which prevent the user by changing them directly, you may be able to avoid such conflicts.

#### Style Guide

~~~~~

Common "style guide" rules (PLEASE NOTE) :

- ControlPads should be located in "ENV:superview-library/" and have the plain name of the SVDriver/SVObject/... plus ".controlpad" as extension.  
For example a ControlPad for JPEG.svobject should be named "ENV:superview-library/JPEG.controlpad".
- for boolean entries use <YES|NO> choices, not <TRUE/FALSE> or just set single Keywords like "USE\_..."
- for switches, <ON|OFF> choices are most often preferable against setting single Keywords for either "on" or "off"
- use pregnant and verbose names for multiple-choices, like <HUFF|LWZ|ENTROPY>, not <1|2|3>
- use short names and statements, not long and complicated ones :  
"ColorDepth=24" instead of "Number\_of\_Colors=16.7\_million"

#### Examples

~~~~~

In "Programmers/C-Language/Example\_Tools/ControlPad" you find some example sources, which deal with ControlPads and should explain anything

you need to know about these.

## 1.105 CPInfo Fileformat

Some applications might wish to enable the user to interactively specify controlpad settings just when reading, writing or processing an image and not before or after by setting preferences.

Until this will be managed by `superview.library`, `superviewsupport.library` or the single modules by introducing new functions for handling this, applications may take this information from ".cpinfo" files, where the specific ControlPad entries are dynamically described.

Content

=====

- Structure in general
- Structure in detail
- Structure Entry Description in detail
- Examples
- Last Words and Exceptions
- Future
- Special configuration Standards

Structure in general

-----

Every CPInfo-File is constructed like an usual ControlPad file, but in this case, the `_order_` of the entries becomes meaningful. These files are separated into SECTIONS, where any necessary information about specific ControlPad entries are stored.

To allow easy enhancements and improvements on this format, there's an important note:

- the number of sections is not limited
- each SECTION `_must_` contain at least the entries described below
- each SECTION `_may_` contain more entries, so that it might be necessary to skip all the following entries until another "SECTION" entry will be reached
- the order of the entries is `_fixed_`
- if any of the following `_needed_` entries makes no sense, it will still be present as a "dummy", thus just a Keyword is placed there and no value is given via "="
- any single line should not exceed 255 characters
- as with usual ControlPad files, comments and blank lines are allowed

Structure in detail

-----

|                                    |                                 |
|------------------------------------|---------------------------------|
| CPINFO                             | ; identification                |
| SECTION=<Name of ControlPad entry> | ; the name of the entry         |
| TYPE=<INTEGER FLOAT ASCII NONE>    | ; type of data                  |
|                                    | ; (signed or unsigned for       |
|                                    | ; INTEGER or FLOAT may          |
|                                    | ; be detected via MIN and MAX). |
|                                    | ; NONE is used for KeyWords.    |

|                                      |                                                                                                                                                                                                              |
|--------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MIN=<Value>                          | ; - minimum value for numbers<br>; - minimum length for strings<br>; - or empty ("MIN")<br>; Empty with KeyWords.                                                                                            |
| MAX=<Value>                          | ; - maximum value for numbers<br>; - maximum length for strings<br>; - or empty ("MAX")<br>; Empty with KeyWords.                                                                                            |
| DEFAULT=<Default value for entry>    | ; what will be used if nothing<br>; is specified ?!<br>; Empty e.g. with KeyWords.                                                                                                                           |
| DESCRIPTION=<descriptive text>       | ; what does it do ?                                                                                                                                                                                          |
| MASK=<input mask>                    | ; this one allows to separate<br>; choices and "free" enterings<br>; Empty with KeyWords.                                                                                                                    |
| WHEN=<READ WRITE ALWAYS PREFS NEVER> | ; ask for it, when the user<br>; does reading or writing<br>; (SVOBJECTS) ? Or always<br>; (SVDIVERS/SVOPERATORS) ?<br>; Or never ? (Undocumented)<br>; Or Just handle it as a real<br>; preferences thing ? |

#### Structure Entry Description in detail

##### o A few words to the MASK entry:

Any entries which don't have a specific meaning (just comments) or mutual exclude entries do have to begin with "<" and end with ">". Anything else are control sequences, which either are to be supported or have to be ignored.

##### a) "<...>" or "<...|...|...>"

Usually you should print out this text as a little help for the user, how to enter the data, e.g. into a string/integer gadget. But if the string included by "<" and ">" does contain one or more "|" these are meant as exclusive choices, which e.g. might be represented as mx-, cycle- or listview-gadgets. So you may a) present this mask to the user while editing a string/integer gadget or b) parse it and just "filter" the input via supplying appropriate input gadgets. Please note, that the "MIN" and "MAX" values (if specified) still have to be valid for this input (if the file definition has been done correctly ;-)

So for a "MASK=<8|24>" these would equal "MIN=8" and "MAX=24". But note, that this "mask" is not really meant to define ranges with gaps, like "anything between 5 and 37, but except the numbers between 33.3 und 36.5" 8-)

##### b) Control Sequences

|                |                                                                                              |
|----------------|----------------------------------------------------------------------------------------------|
| MASK=_SVOBJECT | means, that a file from "LIBS:svoobjects" has to be requested, which matches "#?.svoobject". |
| MASK=_SVDIVER  | means, that a file from "LIBS:svdrivers" has to be requested, which matches                  |

```
MASK=_SVOPERATOR      "#?.svdriver".
                        means, that a file from "LIBS:svoperators"
                        has to be requested, which matches
                        "#?.svoperator".
MASK=_FILE             a file to be selected e.g. fromout a
                        requester (it is supposed, that the user
                        does the right choice).
```

#### Examples

-----

Anentry for a KeyWord might look like this:

```
SECTION=ANYDATATYPES
TYPE=NONE
MIN
MAX
DEFAULT
DESCRIPTION=Switches Datatypes support to ANY
MASK
WHEN=PREFS
```

An entry for a float value:

```
SECTION=SCALE_FACTOR
TYPE=FLOAT
MIN=0.1
MAX=2.0
DEFAULT=1.0
DESCRIPTION=Factor for scaling graphics
MASK=<Value>
WHEN=ALWAYS
```

An entry for an ASCII text:

```
SECTION=PACKMETHOD
TYPE=ASCII
MIN=4
MAX=4
DEFAULT=NUKE
DESCRIPTION=How to pack the written data
MASK=<XPK-Packer>
WHEN=WRITE
```

An example for an ignoreable Keyword:

```
SECTION=DEBUGMODE
TYPE=NONE
MIN
MAX
DEFAULT
DESCRIPTION=Enables debugging mode with Confirm-Requesters
MASK
WHEN=NEVER
```

---

### Last Words and Exceptions

---

So you see, that it's easy to describe any of the existing ControlPad variations this way.

The "WHEN" statement is a help for applications to decide, whether and when to allow changing of specific ControlPad entries.

Applications should either ignore entries of type "WHEN=NEVER" or prepared to handle slightly differing entries, like e.g. TYPE=INTEGER entries with a non-set default value. (An example had been the "AVAILMEM" switch of interim versions of JPEG.svobject - which was both: a keyword and a common setting, but should never be changed during runtime.)

"WHEN=NEVER" is just for options with experimental state or debugging purposes.

### Future

---

In the future, there may be a function in superview.library, which allows passing a controlpad list to it, which then may be passed to the attached SVObject/SVDriver/SVOperator as a "local" setting, which will override the "global" ControlPad files.

So the CPInfo Files will be helpful for applications on how to find out, which ControlPads might be possibly set.

### Special configuration Standards

---

For special cases it seems to be suitable to define a common behaviour, so that GUI-based and interactive selections for those special cases may become independent from specific modules.

#### Applying operations to parts of images only (crop standard)

---

Suggested by Steve Quartly in 11/95

Operators and other modules, which allow to apply their specific operations to either the whole given image or only a defined part of it, should use the following controlpads as a standard for allowing the application to better visualize the process of selecting the concerned parts of the graphics.

```
REGION=<ENTIREIMAGE|RECTANGULAR>
; if this one exists, you have the choice.
; Then, if REGION=ENTIREIMAGE, the following are supposed to be
; 0, 0, width, height, otherwise they have to be specified.
LEFTEDGE=<Value smaller or equal width>
TOPEDGE=<Value smaller or equal height>
WIDTH=<width of shape>
HEIGHT=<height of shape>
```

; just like opening a window

## 1.106 ControlPad Overview

```
=====
MAIN LIBRARIES
-----
ControlPad-Name      : "ENV:SuperView-Library/LIBRARY.controlpad"
ControlPad-Commands : - DEFAULTSVDRIVER=<#?.svdriver>
                      ; Which SVDriver should be set, when
                      ; SuperView-Library is being initialized ?
                      ; For example: "DEFAULTSVDRIVER=AGA.svdriver"
                      - ANYDATATYPES
                      ; if this KeyWord is set, ANY DataTypes will
                      ; be loaded and tried to be displayed in some
                      ; way (e.g. not only pictures, but also 8SVX
                      ; sounds or ANIM-Files ...)
                      ; These changes are recognized each time, when
                      ; a new handle for loading a file is being
                      ; initialized - but may be superseded by specific
                      ; program's settings internally.
-----
ControlPad-Name      : "ENV:SuperView-Library/superviewsupport.controlpad"
ControlPad-Commands : - C2P=<OS|SV>
                      ; determines, whether chunky to planar conversion
                      ; is managed via the appropriate OS functions
                      ; or via internal ones.
                      ; Due to various problems with the OS functions
                      ; "SV" is now default.
                      ; Be careful when using "OS" with GfxCards,
                      ; which carelessly patched graphics.library.
=====
SVOBJECTS
-----
ControlPad-Name      : "ENV:SuperView-Library/AmiFIG.controlpad"
ControlPad-Commands : - AMIFIG_PATH=<fig2dev command path plus name>
                      ; how fig2dev is to be called
                      ; e.g. AMIFIG_PATH=Work:AmiFIG/fig2dev
                      ; default is: fig2dev
                      - STATUS=<ENABLED|DISABLED>
                      ; allows to disable this module - for example
                      ; to be able to use an other, program-specific
                      ; import-module for the same file format
-----
ControlPad-Name      : "ENV:SuperView-Library/EPS.controlpad"
ControlPad-Commands : - EXTRACTMODE=<HEADER|PS>
                      ; Determines, whether the preview image or
                      ; the Postscript (TM) part should be extracted,
                      ; while parsing the Postscript (TM) part requires
                      ; a working Ghostscript installation with
                      ; GhostScript.svobject being correctly configured
                      ; default is: HEADER
                      - SAVE_ROTATE=<ON|OFF>
                      ; by default, the picture seems to be rotated
                      ; with Postscript (TM) output of this module.
```

```

; Enabling SAVE_ROTATE will again re-rotate it to
; the original position.
; default is: OFF
- SAVE_CENTER=<ON|OFF>
; centers the images on the page with the
; desired size of SAVE_WIDTH x SAVE_HEIGHT (or not)
; default is: ON
- SAVE_RLE=<ON|OFF>
; allows to apply RLE compression to the
; PS output
; default is: OFF
- SAVE_DPI=<value>
; DPI value to use for the output
; default is: 300
- SAVE_WIDTH=<value>
; Width of the PS page
; default is: 612
- SAVE_HEIGHT=<value>
; Width of the PS page
; default is: 762

```

---

```

ControlPad-Name      : "ENV:SuperView-Library/FastILBM24.controlpad"
ControlPad-Commands : - BUFFERSIZE=<Size in Bytes>
; how many bytes should be buffered
; (more = faster)
; Default is 8192, minimum is 256
- PASSTHROUGH
; do not recognize ANY pictures, so that
; they e.g. might be passed through to
; ILBM.svobject instead
- DITHERMODE=<HAM6_QUICK|HAM8_QUICK>
; Default is HAM6_QUICK

```

---

```

ControlPad-Name      : "ENV:SuperView-Library/GhostScript.controlpad"
ControlPad-Commands : - GS_PATH=<gs command path plus name>
; how GhostScript is to be called
; e.g. GS_PATH=Ghostscript:gs000
; default is: gs000
- OUTPUTMODE=<PNG256|PNG24BIT>
; Output file format to be used
; (-sDEVICE parameter of GhostScript's gs).
; Only PNG supported yet.
; default is: PNG256
- DPI=<value>
; DPI value to use for the converted output
; (-r parameter of GhostScript's gs)
; default is: 72
- STATUS=<ENABLED|DISABLED>
; allows to disable this module - for example
; to be able to use an other, program-specific
; import-module for the same file format

```

---

```

ControlPad-Name      : "ENV:SuperView-Library/GIF.controlpad"
ControlPad-Commands : - GIF89A_TRANSPARENT_COLOR=<0..255>
; Default color to be enabled and set
; (graphics control extension block)
; Default: (not active)

```

---



---

```

ControlPad-Name      : "ENV:SuperView-Library/GPlot.controlpad"
ControlPad-Commands : - GPlot_PATH=<gplot command path plus name>
                      ; how gplot is to be called
                      ; e.g. GPlot_PATH=Work:GPlot/GPlot
                      ; default is: GPlot
                      - STATUS=<ENABLED|DISABLED>
                      ; allows to disable this module - for example
                      ; to be able to use an other, program-specific
                      ; import-module for the same file format

```

---

```

ControlPad-Name      : "ENV:SuperView-Library/ILBM.controlpad"
ControlPad-Commands : - ANIM_BODIES
                      ; if set, ANIM files' ILBM-BODY chunks
                      ; (first frame) will be extracted when
                      ; such a file is encountered. Otherwise
                      ; ANIM.datatype may do that task later.

```

---

```

ControlPad-Name      : "ENV:SuperView-Library/JPEG.controlpad"
ControlPad-Commands : ;
                      ; STANDARD settings
                      ; *****
                      - COLORDEPTH=<8|24>
                      ; For READING colored 24 bit images:
                      ; should the output be done as 8 or 24 bit Data ?
                      ; default is 8 Bit data. Grayscaled graphics
                      ; will _always_ be exported as 8 Bit
                      - QUANTIZATION=<0..100>
                      ; For WRITING 8/24 bit images as 24 bit JPEG:
                      ; 0 - poor quality, big compression
                      ; 100 - best quality, no compression
                      ; never use such extreme values, but try
                      ; something like 20, 50, 75
                      ; default is: 75)
                      ;
                      ; ADVANCED settings
                      ; *****
                      - FORCE_GRAY_DECODE
                      ; For READING images:
                      ; this switch overrides the selected depth
                      ; for colored images and always exports them
                      ; as 8 Bit grayscaled pictures
                      ; default is: not set
                      - DECODE_METHOD=<FAST_INTEGER|SLOW_INTEGER|
                                FLOATING_POINT|FASTEST_INTEGER>
                      ; For READING images:
                      ; how to DECODE the 24 bit or gray data
                      ; default is: FAST_INTEGER
                      - DITHERMODE=<DITHER_FLOYD-STEINBERG|
                                NO_DITHERING|DITHER_ORDERED>
                      ; For READING and dithering 24 bit images
                      ; to 8 Bit images:
                      ; how and whether to DITHER the 24 bit data
                      ; default is: DITHER_FLOYD-STEINBERG
                      - COLOR_QUANTIZING=<FAST|SLOW>
                      ; For READING and dithering 24 bit images
                      ; to 8 Bit images:

```

---

```

; whether to QUANTIZE the 24 bit data within
; two passes
; default is: SLOW
- UPSAMPLING=<ON|OFF>
; For READING and dithering 24 bit images
; to 8 Bit images:
; whether to do fancy upsampling on the 24 bit
; default is: ON
- FORCE_GRAY_ENCODE
; For WRITING images:
; this switch causes grayscales to be written
; no matter, whether the input was colored
; default is: not set
- ENCODE_METHOD=<FAST_INTEGER|SLOW_INTEGER|
                FLOATING_POINT|FASTEST_INTEGER>
; For WRITING images:
; how to ENCODE the 24 bit or gray data
; default is: FAST_INTEGER
- OPTIMIZE=<ON|OFF>
; For WRITING 24 bit images:
; whether to optimize the generated huffman
; code (good compression, but slow)
; default is: OFF
- PROGRESSIVE=<ON|OFF>
; For WRITING 24 bit images:
; whether to write progressive JPEG files
; default is: OFF

```

---

```

ControlPad-Name      : "ENV:SuperView-Library/Limbo.controlpad"
ControlPad-Commands : - LIMBO_PATH=<Limbo command path plus name>
                       ; how Limbo is to be called
                       ; e.g. LIMBO_PATH=Work:Limbo/Limbo.68030.881
                       ; default is: Limbo.68000
- STATUS=<ENABLED|DISABLED>
                       ; allows to disable this module - for example
                       ; to be able to use an other, program-specific
                       ; import-module for the same file format

```

---

```

ControlPad-Name      : "ENV:SuperView-Library/PCD.controlpad"
ControlPad-Commands : - OUTPUTFORMAT=<BASE/16 | BASE/4 | BASE >
                       ; specifies the output resolution to be used

```

---

```

ControlPad-Name      : "ENV:SuperView-Library/PNM.controlpad"
ControlPad-Commands : - SUPPRESS_HEADER
                       ; setting this keyword will suppress
                       ; writing the PNM header for P5/P6
                       ; files, which actually will generate
                       ; a RAW data file, which cannot be loaded
                       ; with superview.library again, but may be
                       ; used for different purposes
                       ; Default: (not set)

```

---

```

ControlPad-Name      : "ENV:SuperView-Library/Printer.controlpad"
ControlPad-Commands : - DENSITY=<density value for printing quality>
                       ; Amiga's printer device does allow seven
                       ; modes of printing density (1-7)
                       ; default is: 3

```

---

```
-----
ControlPad-Name      : "ENV:SuperView-Library/Scanner.controlpad"
ControlPad-Commands : - PUBSCREEN_NAME=<PubScreenName>
                      ; where the scanner drivers should open
                      ; their windows (if not set or not available,
                      ; the default Public Screen will be used).
-----
```

```
ControlPad-Name      : "ENV:SuperView-Library/SVG.controlpad"
ControlPad-Commands : - PACKMETHOD=<xxxx>
                      ; if this one is specified, it is tried to
                      ; pack the resulting file with the specified
                      ; XPK-Packer. If this fails, the file keeps
                      ; unpacked.
=====
```

#### SVDRIVERS

```
-----
ControlPad-Name      : "ENV:SuperView-Library/AGA.controlpad"
ControlPad-Commands : - 24BITOPERATOR=<OperatorName>
                      ; (case-sensitive, ".svoperator" may be added)
                      ; e.g. "24BITOPERATOR=24BitToHAM.svoperator"
                      ; or   "24BITOPERATOR=ExtractGrayScales"
                      ; specifies, which operation should be performed
                      ; on 24 bit graphics before displaying them
                      ; (if not specified, "best guess" colors will
                      ; be used, which is really slow)
- BITMAPCOPY=<DIRECT|RTG>
  ; "BITMAPCOPY=RTG" prevents AGA.svdriver from
  ; directly copying into Bitmaps, which will
  ; result in a usage of more memory, but keeps it
  ; working.
  ; Default is "BITMAPCOPY=DIRECT".
- SCREENINFRONT
  ; Put Screen to front before the graphics
  ; has been displayed (useful with GfxCards)
-----
```

```
ControlPad-Name      : "ENV:SuperView-Library/ECS.controlpad"
ControlPad-Commands : - 24BITOPERATOR=<OperatorName>
                      ; (case-sensitive, ".svoperator" may be added)
                      ; e.g. "24BITOPERATOR=24BitToHAM.svoperator"
                      ; or   "24BITOPERATOR=ExtractGrayScales"
                      ; specifies, which operation should be performed
                      ; on 24 bit graphics before displaying them
                      ; (if not specified displaying will fail)
- 8BITOPERATOR=<OperatorName>
  ; (case-sensitive, ".svoperator" may be added)
  ; e.g. "8BITOPERATOR=ExtractGrayScales"
  ; specifies, which operation should be performed
  ; on non-ECS graphics (more than 16 Colors in
  ; HighRes, more than 32 Colors in LowRes, HAM8)
  ; (if not specified, will be tried to display)
- BITMAPCOPY=<DIRECT|RTG>
  ; "BITMAPCOPY=RTG" prevents ECS.svdriver from
  ; directly copying into Bitmaps, which will
  ; result in a usage of more memory, but keeps it
  ; working.
  ; Default is "BITMAPCOPY=DIRECT".
- SCREENINFRONT
```

```

; Put Screen to front _before_ the graphics
; has been displayed (useful with GfxCards)

```

```

-----
ControlPad-Name      : "ENV:SuperView-Library/CyberGraphics.controlpad"
ControlPad-Commands : - EMUSCREENDEPTH=<16|24>
                      ; beginning depth for opening CyberGraphics Screens
                      ; Default is 16 Bit (which will also be tried
                      ; when opening of a 24 bit Screen fails)
                      ; This does not concern colordepths < 16 Bit,
                      ; except HAM6/8.
                      - SMALLSCREENS
                      ; unless this KeyWord is specified, it is not
                      ; tried to open screens smaller than 320x240

```

```

-----
ControlPad-Name      : "ENV:SuperView-Library/PicassoII.controlpad"
ControlPad-Commands : - SMALLSCREENS
                      ; unless this KeyWord is specified, it is not
                      ; tried to open screens smaller than 320x240
                      - BLITTER
                      ; uses the Picasso blitter to copy graphics
                      ; into screens. Otherwise CPU is used.
                      ; On 68000 systems you may wish to use the blitter
                      - EMUSCREENDEPTH=<16|24>
                      ; beginning depth for opening Picasso Screens
                      ; Default is 24 bit (if opening fails, it is
                      ; also tried to open a 16 Bit Screen, then)
                      - SCREENMODEREQUEST
                      ; if this one is specified, the SVDriver ITSELF
                      ; will open a ScreenMode-Requester and ask
                      ; for an appropriate Screenmode to use
                      ; Useful, if you always like to change modes.
                      - AUTOSCROLLADJUST
                      ; this keyword will force Autoscroll whenever
                      ; it would make sense, but the Picasso Software
                      ; would not manage it by itself (when either
                      ; only width or height need to be autoscrollled)

```

```

-----
ControlPad-Name      : "ENV:SuperView-Library/Retina.controlpad"
ControlPad-Commands : - EMUSCREENDEPTH=<16|24>
                      ; beginning depth for opening Retina Screens
                      ; Default is 24 bit (if opening fails, it is
                      ; also tried to open a 16 Bit Screen, then)

```

#### SVOPERATORS

```

-----
ControlPad-Name      : "ENV:SuperView-Library/24BitToHAM.controlpad"
ControlPad-Commands : - DITHERMODE=< HAM6_QUICK|HAM6_WELL
                      |HAM8_QUICK|HAM8_WELL>
                      ; specifies the HAM-Mode to be used and
                      ; the resulting speed/quality

```

```

-----
ControlPad-Name      : "ENV:SuperView-Library/CallPNM24.controlpad"
ControlPad-Commands : - PNMCOMMAND=<PBM command path and name>
                      ; PNM command to be called
                      ; e.g. PNMCOMMAND=Work:NetPBM/pnm/pnmscale
                      - PNMOPTIONS=<Options>
                      ; Options for PNM command to be called

```

```

; e.g. PNMOPTIONS=-xscale 2.0 -yscale 2.0
-----
ControlPad-Name      : "ENV:SuperView-Library/Crop.controlpad"
ControlPad-Commands : - CROP_LEFTEDGE=<Value>
                      ; crop from x position
                      ; (will be adjusted, if >= source width)
- CROP_TOPEDGE=<Value>
                      ; crop from y position
                      ; (will be adjusted, if >= source height)
- CROP_WIDTH=<Value>
                      ; crop how many x pixels from leftedge
                      ; (will be adjusted, if too large)
- CROP_HEIGHT=<Value>
                      ; crop how many y pixels from topedge
                      ; (will be adjusted, if too large)
-----
ControlPad-Name      : "ENV:SuperView-Library/Dither24Bit.controlpad"
ControlPad-Commands : - COLORDEPTH=<1..8>
                      ; specifies the colordepth of the dithering
                      ; output (1->2 Colors .. 8->256 Colors)
- DITHERMODE=<BESTPEN|DITHER_FLOYD-STEINBERG
                      |ORDERED|BURKES>
                      ; whether to just select the best pen or
                      ; do Floyd-Steinberg pixel error adjustment
-----
ControlPad-Name      : "ENV:SuperView-Library/ExtractGrayScales.controlpad"
ControlPad-Commands : - COLORDEPTH=<1..8>
                      ; specifies the colordepth of the grayscaled
                      ; output (1->2 Colors .. 8->256 Colors)
- QUICK
                      ; uses >>2, >>1, >>3 (*0.25, *0.5, *0.125)
                      ; instead of *0.3, *0.59, *0.11
-----
ControlPad-Name      : "ENV:SuperView-Library/HilbertDither256.controlpad"
ControlPad-Commands : - BACKGROUND=<BLACK|WHITE>
                      ; defines, which of the two colors will act
                      ; as background color. Useful e.g. for printing.
-----
ControlPad-Name      : "ENV:SuperView-Library/PaletteDither.controlpad"
ControlPad-Commands : - COLORDEPTH=<1..8>
                      ; specifies the colordepth of the final,
                      ; dithered graphics (1..8 for 2..256 colors)
- DITHERMODE=<BESTPEN|DITHER_FLOYD-STEINBERG>
                      ; whether to just select the best pen or
                      ; do Floyd-Steinberg pixel error adjustment
- PALETTEDEPTH=<1..8>
                      ; number of colors (depth) to be taken from the
                      ; fixed palette - the possible rest will be
                      ; generated (if specifying more palette colors
                      ; than available, all available will be taken)
- PALETTEFILE=<palette graphics filename>
                      ; any 2..256 color graphics file, of which
                      ; superview.library is able to extract a
                      ; palette from (for example an IFF-ILBM file,
                      ; but including a BMHD and - maybe empty - BODY).
                      ; Allows to take one graphics as sample for
                      ; the others (concerning the palette).

```

```
ControlPad-Name      : "ENV:SuperView-Library/Rotate.controlpad"
ControlPad-Commands  : - DEGREES=<90|180|270>
                        ; rotate by how many degrees (reverse clockwise) ?
```

```
ControlPad-Name      : "ENV:SuperView-Library/RotateFree.controlpad"
ControlPad-Commands : - ROTATE_ANGLE=<0..360>
                      ; rotate by how many degrees (reverse clockwise) ?
                      - METHOD=<MIDDLE|GIVENPOINT>
                      ; default is rotation about the middle
                      - X_COORD=<Value>
                      ; if METHOD=GIVENPOINT we need (X/Y)
                      ; Default is (0/0)
                      - Y_COORD=<Value>
                      ; if METHOD=GIVENPOINT we need (X/Y)
                      ; Default is (0/0)
                      - KEEPSIZE
                      ; if this keyword is set, it is NOT tried to
                      ; readjust the image size and to center the
                      ; image, so that parts, which would be out of
                      ; range won't be cut off (which works best
                      ; with (X/Y) pairs in the left, upper quarter)
```

```
ControlPad-Name      : "ENV:SuperView-Library/Scale50.controlpad"
ControlPad-Commands  : - METHOD=<HALF|DOUBLE>
                      ; scale to which size ?
```

```
ControlPad-Name      : "ENV:SuperView-Library/XOR.controlpad"
ControlPad-Commands  : - XORVALUE = <0..255>
                        ; 0 and 255 will not be the best decision ;-)
```