SuperView-Library

COLLABORATORS			
	TITLE : SuperView-Library		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		July 22, 2024	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

SuperView-Library

1.1 SuperView Library Documentation

superview.library V10.2

- FREEWARE -

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

Release Date : 17.2.1995

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

1.2 Copyright

The superview.library and the distributed files (e.g. the documentation files) are (C)opyright 1993-95 by Andreas R. Kleinert. All rights reserved.

(For some files there might additional or substitutive Copyrights take place, which are stated locally within the documentation, then.)

The usage of the "superview.library"'s functions for own program projects, no matter if commercial or non-commercial, is allowed without any restrictions.

Some of the mentioned names or products may be copyrighted by companies or trademarks of companies.

Remark: Some (Drawer-)icons have been taken from Fred Fish's
        AmigaLibDisks.

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.

1.4 Distribution

Non-commercial distribution and usage

The program SuperView-Library in this version is freely distributable.
You may copy it, if the copyright notice is left intact and all of its parts are included in the distribution.

The program may be put on public domain disks or included in public domain disk libraries.

Special permission hereby goes to Fred Fish's AmigaLib-Disks and the german series (in alphabetical order) :
AmigaSzene, BerndsPD, FRANZ, GPD, SaarAG, TAIFUN and TIME.

This program may also be distributed via electronic mail and may be put into mailboxes as long as the redistribution conditions are respected in all points.

The program must not be sold in any way, but it is allowed to take a nominal fee including the costs for copying.

Programmers of freely distributable programs, which make use of superview.library, may create their own distributions of superview-library for inclusion with their programs, but they must state within the program's documentation from where the whole and original distribution can be obtained (e.g. from AmiNet).

I would like to get a copy of the final program, if possible (or at least an EMail reporting that such exists).

Commercial usage

This program MUST NOT be included in commercial packages or commercial program collections without my written permission, e.g. expressed in a license agreement.

Nevertheless programmers of commercial programs may integrate support of superview.library into their programs and make full use of its facilities.

So there are no restrictions in using, programming and supporting it, but no commercial distribution (e.g. shipping the library together with commercial programs) is allowed.
Also, in my opinion, there's no difference in selling freely distributable software together with your product or instead directly including it with the package. A license will be needed for this.

Also I demand to get a copy of the final commercial program directly from the company or one of its subsidiaries.

Contact me for detailed information on license agreements.
The fees are moderate and will not substantially decrease your effective profit too much. Last not least it is the best solution to please your customers.

... and REMEMBER :

By using or distributing this program you automatically agree to all of the above conditions and terms.

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 (IFF-ILBM, GIF, JPEG, PCX, Targa, TIFF, ...).

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

Programmers may send me their own SVObjects, SVDrivers or SVOperators, if they want to have them added to the main distribution archive or put onto the DeveloperDisk, which is available for the library.

1.6 Installation and Configuration

Just use the supplied Installer-Script, or "by hand"-copy "superview.library" and "superviewsupport.library" to your "LIBS:"-Directory, the SVObjects to "LIBS:svobjects/", the SVDrivers to "LIBS:svdrivers/" and the SVOperators to "LIBS:svoperators/" !

"datatypes.library" V39+ is only needed, if OS3-DataTypes are wished to be supported.
Please take care, which libraries are also additionally needed by the single SVObjects/Drivers/SVOperators !!!

"AddBuffers"-Values for the used Drives should at least be around 25, sometimes more might be useful.

Assignments

As always, there have to be the standard assignments to be present like "ENV:", "ENVARC:" and so on

Addionally, 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) or JPEG.svobject (jpeg_XXX).
It does not always make sense, to place this directory somewhere on 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.

Preferences

You may set and/or modify the following ControlPad-Switches for use with the superview.library.
Please also note, that many SVObjects/Drivers/Operators may have their own ControlPad-Support.
Default-values will be used, if these files and/or settings are not available.

ControlPad-Switches

```
ControlPad-Name     : "ENV:superview-library/LIBRARY.controlpad"
ControlPad-Commands : - 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 ...)
                      - DEFAULTSVDRIVER=<#?.svdriver>
```

```
                              ; Define, which SVDriver should be used
                              ; by default.
                              ; E.g. "DEFAULTSVDRIVER=AGA.svdriver"
```

1.7 Overview of currently available SVObjects

Available SVObjects (as far as known) :

SVObject	Type	Read-Support	Write-Support
IFF-ILBM	EXT	(system-dependent)+24	(system-dependent)+24
IFF-ACBM	EXT	(system-dependent)	(system-dependent)
OS3-DataTypes	INT	all Pictures	(not available)
PCX upto V3.0	EXT	max. 8/24bit Files	max. 8/24bit Files
SVO upto V1.0	EXT	(system-dependent)+24	(system-dependent)+24
GIF upto 89a	EXT	max 8bit Files	max 8bit Files
BMP (W*nd*ws)	EXT	max 8/24bit Files	max 8/24bit Files
Icon (W*nd*ws)	EXT	16 Color Icons	(not available)
FBM (*nix)	EXT	max 8/24bit Files	max 8/24bit Files
PNM (*nix)	EXT	max 8/24bit Files	always as 24bit Files
C64 (Koala, Doodle)	EXT	max 4bit Files	(not yet available)
IMG (GEM/Ventura)	EXT	max 8bit Files	(not yet available)
TIFF (V5.0)	EXT	max 8/24bit Files	max 8bit Files
EPS	EXT	(various)	(not yet available)
Targa	EXT	max 8/24bit Files	max 8/24bit Files
WPG (WordPerfect)	EXT	max 8bit Files	(not yet available)
SunRaster (RAS)	EXT	max 8/24bit Files	max 8/24bit Files
Pictor/PCPaint/PIC	EXT	max 8bit Files	(not yet available)
MAC (MacPaint)	EXT	max 1bit Files	(not yet available)
IFF-YUVN	EXT	24bit Files	always as 24bit Files
JPEG (JFIF)	EXT	max 8/24bit Files	always as 24bit Files
PCD	EXT	24bit Files	(not available)

There’s also an "Unpack.svobject" in the Bonus-Directory, which allows reading of packed files without explicitely unpacking them before (supports XPK and PP20).

Planned SVObjects (no guarantee !) :

- RIFF
- MTV
- SGI
- RLE
- Imagine Textures (if possible)
- RGB8/RGB24
- 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)

1.8 Overview of currently available SVDrivers

Available SVDrivers (as far as known) :

SVDriver	Requirements	BITPLANE	ONEPLANE
ECS	ECS*, OS V2.04+ (V37), 68000+	(system)	8bit
AGA	AGA*, OS V3.00+ (V39), 68020+	(system)	8/24bit
EGS-System	EGS-Graphic-Libraries	8bit	8/24bit
OpalVision	OpalVision Card	8bit	8/24bit

* ECS- and AGA.svdriver both should work with Graphic Cards, if there's a Workbench Emulation and/or ScreenMode Catcher available.
If you own a Graphics Card, without having an appropriate SVDriver for it, always try using AGA.svdriver at first.

Planned SVDrivers (no guarantee !) :

- Piccolo/EGS, PicassoII, RetinaZ2/Z3, Merlin-II, CyberVision, etc.

```
*** PicassoII and EGS7 support are expected to be available soon ***
*** Either on AmiNet from third party authors or within one of   ***
*** the next official release of the library                     ***
```

1.9 Overview of currently available SVOperators

Available SVOperators (as far as known) :

SVOperator	
XOR	nice effects ...
24BitToHAM	dithers 24 Bit RGB to HAM6/HAM8
Dither24Bit	dithers 24 Bit RGB to 2..256 Colors
AnyTo24Bit	converts input to 24 Bit
HilbertDither256	dithers to Black & White
ExtractGrayScales	converts (upto) 256 Colors or 24 Bit to Gray
ExtractRed	extracts Red values from (upto) 256 Colors or 24 Bit
ExtractGreen	extracts Green values from (upto) 256 Colors or 24 Bit
ExtractBlue	extracts Blue values from (upto) 256 Colors or 24 Bit
TopToBottom	turns top to bottom ...
LeftToRight	turns left to right ...
Rotate	rotates by 90/180/270 degrees ...
Scale50	scales to half/double size

Planned SVOperators (no guarantee !) :

- HAM6/8 -> 2..256 Colors
  (currently you will have to convert them to 24 Bit before)
- some more operators for common image processing
- and more

Maybe, some of the currently included SVOperators may be merged to more integrated modules (e.g. 24 Bit dithering).

1.10 Software supporting SuperView.library

```
    The following programs are currently supporting the "superview.library" :

     - SuperView
     - SimpleView (superview.library Example Program)
     - MicroView (superview.library Example Program)
     - KFracPlus
     - SIP (for examinations)
     - DRAFU (unreleased Beta-Version; ask for a demo)
     - SuperLoader (Steve Quartly's OpalVision-Loader)
     - SqOpal
     - ...

    There are several more programs (Shareware, commercial), but since
    they are not yet available I don't list them here.
```

1.11 Thanks, greetings, credits and the rest

```
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 used to design the GUI of
   SuperViewPrefs. Thanks!

 * 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 new AmigaGuide 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 I only use the InstallerScript-Icon so far.

 * Jürgen Schäfer

   ... for speficic 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 more Icons for use with SuperView. Additionally, he had some ideas and suggestions on 24 Bit ILBM and YUVN.

- and last NOT LEAST

  - all _registered_ users of SUPERVIEW for supporting Shareware !!
    (ORDER YOUR KEYFILE NOW !)

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

Alex	Carbin	TuC / Co-Sysop Century
Sven	Drieling	Indy - Creator of "PowerBrei" DiskMag
Oliver	Eichhorn	uses the library, made bug-reports + suggestions
Thomas	Fischbach	Sysop of "GM-Box", my current Fido-Boss
Fred	Fish	AmigaLibDisks
Stefan	Grad	GPD-Disks
Klaus	Holtorf	for several information on graphic file formats
Paul	Huxham	(Co-)Programmer of OpalVision-SVDriver
Stefan	Kremer	TuC / Sysop Century
Alex	Lange	Time PD-Disks
H.P.	Lattka	Franz PD-Disks
Andreas	Manewaldt	Taifun PD-Disks
Axel	Melzener	Game Object Design
Andreas	Neumann	Creator of the PCQ-Support includes and modules.
Michael	Petrikowski	Amiga Szene PD-Disks
Albi	Rebmann	Operator of my favoured Fido-InterNet-Gateway, supplier of my InterNet address
Thomas	Saenger	Sysop of "elephant BBS"(down), my former Fido-Boss
Michael	Trautmann	PC-Programmer, for information on WinIcons.
Steve	Quartly	(Co-)Programmer of OpalVision-SVDriver, also wrote "SuperLoader" and "SqOpal".
NJ	Verenini	The one, who created this nice "Spumoni's Workbench" JPEG-Picture (AmiNet) with SuperView ... :-)

  - 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	
Rudi	Brand	
Aaron	Digulla	
Thomas	Gundlach	
Richard	Hartmann	
Jim & Becky	Maciorowski	(thanks for the nice card :-)
Patrick	Ohly	
Jürgen	Schneider	
Reinhard	Theling	
Tilo	Winkler	(thanks for the strange TIFFs ;-)

    and more.

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

1.12 How to contact the author

If you like, you may send me some money. Perhaps this will motivate me to continue programming such programs or just making updates of this one. Send bug-reports, money or whatever to :

Andreas R. Kleinert,
Grube Hohe Grethe 23,
D-57074 Siegen,
Germany.

```
Phone:  +49-271-331859 (weekdays after 18.00h)
        +49-271-332147 (weekdays after 18.00h)
EMail:  Fido             Andreas Kleinert 2:2457/435.10
        Usenet/InterNet  Andreas_Kleinert@superview.ftn.sub.org

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

          Andreas_Kleinert@p10.f435.n2457.z2.fido.sub.org (in Germany)
          Andreas_Kleinert@p10.f435.n2457.z2.fidonet.org  (USA or other)
```

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)/1300/2000/ 2500(UX)/3000(T)/4000(/030/040)(T), DraCo, Kickstart/OS Version, RAM, HardDisk, GfxCard, any special Hardware/Software).

1.13 The future of the SuperView.library

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

Well, ideas cannot be planed 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 fincancial support, but also additional programming efforts like third-party improvements on the library.

1.14 History

Please note the version-dependencies :

superview.library	SVObjects	SVDrivers	SVOperators
Version 1	-	-	-
Version 2	Version 1	-	-
Version 3-8	Version 1,2	Version 1	-
Version 9-10	Version 1,2	Version 1	Version 1

V10.2 (17.2.1995) :

- Bug-Fixes:
  - ECS.svdriver, AGA.svdriver:
    If a window couldn't be opened, the screen will now be closed immediately (-> Patrick Ohly)
    This would have been done within SVD_FreeHandle() sooner or later, but this way it is smarter ;-)
  - Drivers, main libraries: fixed behaviour on library open errors when initializing (before, there might have been some libraries, which would not have been closed on an error) (-> Patrick Ohly)
  - ILBM/ACBM.svobject:
    page dimensions no longer influence gfxbuffer size (-> Patrick Ohly, Jürgen Schäfer)
  - 24BitToHAM.svoperator produced "HAM6_WELL", when "HAM8_QUICK" was specified and vice versa. Fixed.
  - InstallerScript no longer sets obsolete "Default.svdriver", but new "DEFAULTSVDRIVER=..." in LIBRARY.controlpad. Note: will overwrite old file completely, so that you might skip this part in expert mode.
  - fixed SampleConfigs a little ;-)
- New features:
  - added "AnyTo24Bit.svoperator", which transforms upto 256 Colors and HAM6/HAM8 into 24 Bit.
  - added locale-support for OS V38+ (error texts).
    also changed InstallerScript to handle installation.
- Documentation and Programmers Section:
  - added ControlPad-FAQ ;-)

V10.1 (2.2.1995) :

- New features:
  - added SVOperator-Support for 24 Bit graphics to ECS.svdriver (just similar to AGA.svdriver)
  - "ExtractGrayScales.svoperator":
    added ControlPad-Support for creating grayscaled graphics with less than input colors (greyscales); even 24 Bit may now be scaled to less than 256 grayscales
  - "Targa.svobject": added support for RLE-compressed files (reading)
  - Datatypes may now directly be extracted as SV_GfxBuffers.
    This works as well with Anims (first picture) and other Datatypes, if ANYDATATYPES is specified.
    Displaying of non-picture Datatypes is either possible interactively (e.g. Play-Button for sounds or anims when calling SuperView's "View File") or as a plain, single-picture GfxBuffer (e.g. when calling SuperView's "Load" and "View").

```
         The Buffer is created internally, when calling ReadToGfxBuffer()
         and will be used, if necessary, otherwise not.
         This changes behaviour of the library, but is still compatible
         (side-effect: picture-datatypes may now be scrolled just like
          usual pictures, 'cause they're now used as GfxBuffers).
          May need more memory, sometimes.
     - Bug-Fixes:
       - "BMP.svobject": did not set correct ViewMode for 24 Bit graphics
     - Misc:
       - added some sample configurations to "Docs/SampleConfigs".
         One for ECS, one for AGA/GfxCards. Copy them to "env:superview-
         library"
       - new InstallerScript-Icon (-> Roman Patzner, see given credits)


     V9.13 (28.1.1995) :
     -------------------
     - Note: There have been several pre-release and beta-versions
             of several SV-Modules and the library itself.
             Some were (more or less) official, some not.
             Do not use them any longer. This is the final release of V9.13.
     - New features:
       - added "EPS.svobject", which passes the Trailers/Previews/Thumbnails
         of EPS graphics (usually TIFFs) through superview.library.
       - added "PCD.svobject", as a "Bonus" add-on
         (read the documentation before using it!)
       - added "Rotate.svoperator" for 90/180/270 degrees (reverse clockwise)
       - added "Scale50.svoperator" which scales any graphics by 50%
         (half/double size)
       - modified "24BitToHam6.svoperator" to create HAM8 output,
         if requested, also added "quick" options for both, HAM6 and
         HAM8. Renamed it to "24BitToHAM.svoperator".
     - Bug-Fixes:
       - "YUVN.svobject": fixed YUV decoding + encoding
       - "BMP.svobject" : fixed 24 Bit reading/writing: what we used,
                          were RGB-Pixels, but BMP uses BGR-Pixels (oops).
       - "ILBM.svobject": fixed another 24 Bit alignment problem, occured
                          when reading, only (guess, this was the last one)
                          and some other minor things.
       - "PCX.svobject" : several: see there.
       - "superviewsupport.library" contained some ControlPad-Bugs.
     - Changes:
       - "Default.svdriver" variable is becoming obsolete.
         Finally improved handling (no longer complains about extra
         spaces, etc.) but use the new ControlPad of LIBRARY.controlpad
         instead now: "DEFAULTSVDRIVER=#?.svdriver"
         Fixed SuperViewPrefs to also handle this correctly.
     - Documentation and Programmers Section:
       - again: worked on the documentation (removed obsolete and
         redundant information) so that about 30K diskspace finally
         could be saved within the GuideFile :-)
       - updated "thanks"-list ;-)
       - removed some garbage from several Include-Files, also removed
         any links to the planned (but never implemented) feature of
         "independent" SVObjectNode entries. May be re-introduced
         later in a different way, under a different name.
       - please also take a look into the (revised) NotesAndHints.guide
```

```
  - Email address: "superview.life.sub.org" now finally changed
                   to "superview.ftn.sub.org".


V9.12 (29.12.1994) :
--------------------
- New features:
  - added "Dither24Bit.svoperator" for quick and easy dithering
    of 24 Bit RGB data to 2..256 Colors (default is 256).
    May e.g. be used with AGA.svdriver to dither graphics before
    displaying them.
  - added "Unpack.svobject" to Bonus-Directory, which allows to
    parse PP20- or XPK-packed files through superview.library,
    without doing any further actions on them before (writes
    temporary files to VMEM:)
  - added "PNM.svobject" for reading (PBM,PGM,PPM) and writing (PPM)
    of binary PNM-Files, as e.g. used by the (Net)PBM package.
    Supports binary (P4/P5/P6), but not yet ASCII (P1/P2/P3) data.
  - added 24 Bit support to "FBM.svobject" (reading and writing)
- Bug-Fixes:
  - "SunRaster.svobject": 24 Bit files had the ColorMap flag set and
                          contained wrong planetype flag. Fixed.
  - fixed bug in "FBM.svobject" ("bits" value might not have been
    "8" always -> misinterpreted as "depth").
  - fixed bug in the (re-)initialization module of all SVDrivers
    and SVOperators (except OPAL), which might cause a small
    memory loss, when re-loading the specific module after it has
    been flushed out of memory, while superview.library wasn't flushed.


V9.10/9.11 (18.12.1994) :
-------------------------
- Bug-Fixes in V9.11:
  - TIFF.svobject: there always popped up some requesters
                   with debugging info in V9.10
- New features:
  - added SVOperator-Support for 24 Bit graphics to AGA.svdriver:
    via a ControlPad you may now specify, what should be done _before_
    displaying 24 Bit graphics (e.g. calling the new "24BitToHam6"-
    Operator. See notes below and docs of AGA.svdriver).
  - added "24BitToHam6.svoperator" to allow (quick and easy) conversion
    of 24 Bit RGB data to HAM6.
  - PCX.svobject: added write support for 24 Bit graphics (unencoded)

- Bug-Fixes:
  - PCX.svobject: (several)

- Misc:
   - added several new Icons, which have been designed by
     Detlef Winkler. Thanks !
     (See SuperView documentation files, if you wish to contact him.)
   - designed better GUI and Icon for SuperViewPrefs

- Documentation and Programmers Section:
   - added some more details to the distribution/license conditions:
     not actually changed, but precised.
   - revised and updated "requirements" section of this documentation
   - fixed some bugs inside the function reference docs (programmers),
     where some function names and an offset had been wrong
     (-> Andreas Neumann)
   - revised Includes slightly
   - included PCQ-Pascal support-stuff to the Programmers part
     of the library distribution.
     Contains Include-Files and two Example-Sourcecodes (based on
     superview-lib 9.x and superviewsupport-lib 4.x includes).
     Many thanks to Andreas Neumann (NEUDELSoft), who created/edited
     these Files and Programs !
   - removed revision history of versions before V9.1 because of
     diskspace reasons ... ;-)
     See Docs of any older version, if you’re interested in it.
   - JPEG.svobject still came with the old doc-file (V2.4).
     Now V2.5-Docs are included, describing the (small) changes.
   - "SimpleOP.c" example program was neither correct nor useful
     (just debugging code). Now you may take a look at it...
   - fixed bug within programmer’s documentation:
     SuperWrite has only ONE parameter, not TWO !
     (This bug happened while transfering some paragraphs from
      superplay-lib’s documentation to superview-lib’s Docs... ;-)


V9.9 (20.11.1994) :
-------------------
- improved AGA.svdriver a little bit (OS 3.1)
- added WinIcon.svobject for W*nd*ws (TM) Icons, although
  I do not really know, whether anyone might need this...
- added "VIRTUAL MEMORY" support functions to superviewsupport.library.
  Use them as a programmer, or if you’re a user, just wait, until
  they’ll be used by superview-lib...
- fixed bug in BMP.svobject, where the buffer of a written graphics
  would have been turned upsidedown after the operation
  (-> Detlef Winter)
- there were still alignment bugs inside ILBM24-Support
  (-> Detlef Winkler, Steve Quartly)
- recompiled JPEG.svobject and changed and optimized some parts.
  Note, that temporary files no longer will put in "JPEGTMP:",
  but in "VMEM:"


V9.8 (04.11.1994) :
-------------------
- argh!
  Not again!
  SVOperators still did not have their own directories and would
  have been copied (one after an other) to a file called
  "libs:svoperators".
  Now finally fixed (SORRY!)
- added SVOperators for extraction of RED, GREEN and BLUE
  values from (upto) 256 Color graphics or 24 Bit graphics
  (with 256 Color graphics as result).
  Those resulting graphics may easily be converted to 256 Color
  grayscale graphics or simply be used for a "quick view" on
  24 Bit graphics.
  -> ExtractRed, ExtractGreen, ExtractBlue
- replaced "256ToGray.svoperator" with "ExtractGrayScales.svoperator",
  which now will also work with 24 Bit sources.
- updated InterNet-address list


V9.7 (29.10.1994) :
-------------------

- added some functions to superviewsupport.library
  (-> Steve Quartly)


V9.6 (28.10.1994) :
-------------------

- library startup-code: SVDriver and SVOperator lists had not
  been freed (small memory loss, but not significant and only,
  when expunging the library without reboot)
- programmers: "svobjects.fd" was "svoperators.fd" and vice versa.
  Fixed.
- SVL_FileInfoRequest() now also will inform about plain SV_GfxBuffers,
  which resulted out of an SVOperator action ("virtual graphics").
  The results e.g. can be seen when dithering a picture via SuperView
  and then calling "FileInfo": now there’s information available,
  and you’ll see, that the ColorDepth will have changed.
- Installer-Script did not install the SVOperators.
  Users had to do this separately. Sorry: has been fixed.
- ILBM.svobject did no longer write from BitPlanes, only from
  ChunkyPixel-Buffers (aaaaahhhhh, not AGAIN ...)


V9.5 (21.10.1994) :
-------------------

- AGA.svdriver V1.11 had a small (?) bug: autoscrolling of large
  pictures had been disabled while fixing several other things
  (forget to re-activate) it. Sorry.
  Now it will work again (BTW: ECS.svdriver did not have this bug).


V9.4 (15.10.1994) :
-------------------

 - included new YUVN.svobject (IFF-YUVN).
   Already with full write-support (+ clipboard reading/writing).
 - improved SVOperators:    - HilbertDither256
 - fixed bugs in SVObjects: - ILBM: 24 Bit now really works with
                                  all solutions and all destination
                                  programs (hopefully ;-)
 - misc: * note, that one of my InterNet addresses has been removed
           from the list (no longer valid)
 - and more ?!


V9.3 (09.10.1994) :
-------------------

 - aargh: there was no "svoperators"-directory in the archive’s
```

```
               "libs"-path, so that XOR.svoperator had been copied/renamed
               to the FILE "svoperators".
        I'm awfully sorry about this !
        (Thanks to Sven Driehling for reporting this !!)
      - fixed some bugs within the documentation and elsewhere
        (-> Sven Driehling)
      - added missing FD-File (and entries within the other files)
        (-> Sven Driehling)
      - added SVOperators :  - "HilbertDither256" for fractal Hilbert
                               dithering from 256 Colors to 2 Colors
                               Black & White
                             - "256ToGray" for converting 256 Colors
                               to Gray (colormap operation)
      - note: be careful when using the IFF24 Bit support.
        Problems have been reported when converting from Targa (or similar)
        to 24 Bit IFF. Watch out for a bug-fixed version.
        (-> Detlef Winter)


     V9.2 (02.10.1994) :
     -------------------

      - added 24 Bit-Support for IFF-ILBM (reading and writing),
        which has been suggested by several people.
        (requested by Detlev Winkler)
      - small bug-fix inside BMP.svobject
        (mentioned by Gerd Frank)
      - added ControlPad-Switch (ENV:SuperView-Library/LIBRARY.controlpad)
        to allow (again) displaying DataTypes other than for GfxFormats
        (set ANYDATATYPES as switch).
        This is only for people who like to "display" sounds by putting
        the "play sound" button onto a Screen ... ;-)
        This option is disabled by default, so that DataTypes other than
        of kind "picture" will be rejected usually.
        (requested by Gerd Frank)


     V9.1 (30.09.1994) :
     -------------------

      - added SVOperator-Support (and some new functions)
      - fixed some very bad bugs concerning Oneplane/Chunky-Pixel support
      - slightly changed (cleaned up) distribution conditions,
        to prevent future misunderstandings

   *************************************************************
   * Revison history entries before V9.1 have been deleted     *
   *                     to save diskspace !                   *
   *************************************************************
```

1.15 ILBM.svobject

```
Version      : 2.17
Release Date : 15.02.1995

Description
~~~~~~~~~~~
ILBM.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and
needs any SVDriver with Bitplane-Support.

It supports reading and writing of IFF-ILBM Files.
In detail these are :

Reading :

 - IFF-ILBM uncompressed            1..24 Bit
 - IFF-ILBM CmpByteRun1 compressed  1..24 Bit

Writing :

 - IFF-ILBM uncompressed            1..24 Bit
 - IFF-ILBM CmpByteRun1 compressed  1..24 Bit

Reading/writing from/to ClipBoard is supported.

History
~~~~~~~
V2.17 (15.02.1995) :

 - page dimensions no longer influence gfxbuffer size
   (-> Patrick Ohly, Jürgen Schäfer)

V2.16 (22.01.1995) :

 - fixed another 24 Bit alignment problem, which occured, when width
   was not divideable by 8 (reading only, already fixed when writing)
   Example: 767x431 (-> J. Schäfer)
 - fixed some minor problems:
    - improved ScreenMode selection for 24 Bit Graphics without own
      CAMG chunk
    - no longer creation of BITPLANE GfxBuffers with 24 Bit will
      be possible in low-memory situations
    - improved some old code


V2.15 (16.11.1994) :

 - there were still alignment bugs inside ILBM24-Support:
   one occured with both - CmpByteRun1 and umcompressed - and
   one only with uncompressed pictures

V2.14 (28.10.1994) :

 - since there was a special case for 24 Bit Buffers, no longer
   simple BitPlane-Buffers (only ChunkyPixel) would have been
   accepted (error_action_not_supported).
   Fixed.
```

```
  V2.13 (15.10.1994) :

   - bug-fixed 24 Bit-Support: - some programs (V***T**) do not like it,
                                 when the CAMG-ScreenMode is invalid,
                                 but not HAM. Now we do not write any
                                 CAMG-Chunks with 24 Bit-Files, because
                                 even HAM8 would not be the "right one"
                                 for 24 Bit (just for _dithered_ 24 Bit).
                               - odd-width pictures would not have been
                                 read/written correctly.
                                 Simple bit-shifting bug (forgot to add 7
                                 before dividing through 8). Fixed.

****************************************************************************
* Revison history entries before V2.13 have been deleted for space reasons *
****************************************************************************
```

1.16 ACBM.svobject

```
  © 1994-95 by Andreas R. Kleinert.
  FREEWARE. All rights reserved.

  Version      : 2.12
  Release Date : 15.02.1995

  Description
  ~~~~~~~~~~~
  ACBM.svobject is an external Library-Module for the superview.library V2+.

  It contains SVDriver-Support for the superview.library V3+, and
  needs any SVDriver with Bitplane-Support.

  It supports reading and writing of IFF-ACBM Files.
  In detail these are :

  Reading :

   - IFF-ACBM uncompressed

   (except 24 Bit-ACBMs)

  Writing :

   - IFF-ACBM uncompressed

   (except 24 Bit-ACBMs)

  Reading/writing from/to ClipBoard is supported.

  History
  ~~~~~~~
  V2.12 (15.02.1995) :

   - page dimensions no longer influence gfxbuffer size
```

```
     (-> Patrick Ohly, Jürgen Schäfer)

  V2.11 (12.06.1994) :

   - now GfxBuffer-writing no longer results in wrong x/yAspect values

****************************************************************************
* Revison history entries before V2.11 have been deleted for space reasons *
****************************************************************************
```

1.17 Datatypes support

The library optionally accesses OS3-Datatypes.
Usually only the Datatypes for pictures are supported.
This means, no Samples, Text, Amigaguide File or anything else will be displayed via the SuperView.Library (see configuration section for information on how to bypass this) !

1.18 PCX.svobject

```
  © 1994-95 by Andreas R. Kleinert.
  FREEWARE. All rights reserved.

  Version      : 2.16
  Release Date : 27.01.1995

  Description
  ~~~~~~~~~~~
  PCX.svobject is an external Library-Module for the superview.library V2+.

  It contains SVDriver-Support for the superview.library V3+, and
  needs any SVDriver with Bitplane-/ChunkyPixel-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 - no matter, which WriteSubType
   has been specified - 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.

Remarks

- Why writing only 16/256 Color files ?

  When loading 4 Color PCX files, some programs might expect a CGA style color palette, while we only write EGA/VGA palettes. For 8 Color graphics it is not clear how they should be handled.
  Also, most programs do not support 32 .. 128 Color graphics, perhaps because it is not clear, how the colormap has to be stored :
  all at the end of the file or one hunk in the header and only the rest at the end of the file ?

  So why should PCX.svobject support things, which are not defined as clear as they should be ?

- Then, why reading 4/8 Color files nevertheless ?

  Because PCX.svobject wrote them - this was my fault - before I fixed this bug. So it still reads them, but no longer writes them.

- What about PCC-Brush-Files ?

  These should also be supported, because they’re just a variation of PCX-Files - usually only the extension is changed - but it seems to me, that some of them are not correctly written, so that reading may sometimes result in garbage.

- What about 24 Bit-Files :

  Reading of RLE-encoded files works. It has been tested.
  Reading of uncompressed files should now work, due to the bug-fix in V2.11, although it has not been tested yet.
  Send me your Public Domain 24 Bit Pictures, if they do not work with PCX.svobject yet !

History

V2.16 (27.01.1995) :

- we did compute "Bytes Per Line" by ourself, which sometimes caused problems with odd-width graphics or pictures which use a bpl-value, which isn’t directly related to the width (e.g. padded to longword-boundaries). Fixed now for RLE-encoded files with 1-n, 8 and 24 Bits. Also 8 and 24 Bit routines now accept "compressed across lines" graphics.
- expect more bug-fixes for a) uncompressed and b) rather strange

```
     PCC-brush files

  V2.15 (11.12.1994) :

   - simplified selection of PCX output format, since this never
     has been supported, anyway (now: "PCX V2.5-3.00")
   - removed debugging message, which occured with Bitplane data sometimes
     (also removed various debugging code)
   - Oneplane data with upto 16 Colors could not be written. Fixed.
   - added support for writing 24 Bit graphics (unencoded)

  V2.14 (09.09.1994) :

   - fixed bug in reading one of the many PCX derivates :
     files, which contain RLE-codes which compress across lines
     (some planar files with colors <= 16) now will be read correctly
     (no longer garbage on the screen or "memory corrupt").
     Algorithm had to be changed slightly to allow universal handling,
     but should not be slowed down significantly (because also some
     optimizations have been done ;-)
   - also _some_ PCC-brushes _might_ now be read (better) ?!
     (don't know whether those, which work _now_ also did work _before_
      ... ;-))

****************************************************************************
* Revison history entries before V2.14 have been deleted for space reasons *
****************************************************************************
```

1.19 SVO.svobject

```
  © 1994-95 by Andreas R. Kleinert.
  FREEWARE. All rights reserved.

  Version      : 2.8
  Release Date : 01.07.1994

  Description
  ~~~~~~~~~~~
  SVO.svobject is an external Library-Module for the superview.library V2+.

  It contains SVDriver-Support for the superview.library V3+, and
  needs any SVDriver with Bitplane-/ChunkyPixel-Support.

  It supports reading and writing of SVO Files.
  In detail these are :

  Reading :

   - SVO V1.0 ONEPLANE/BITPLANE (includes 24 Bit Files)


  Writing :

   - SVO V1.0 ONEPLANE/BITPLANE (includes 24 Bit Files)
```

```
ControlPad-Switches
~~~~~~~~~~~~~~~~~~~
ControlPad-Name     : "ENV:superview-library/SVO.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
~~~~~~~~~~~~~~~~~~~~~~~
Pictures in the "SVO Graphics File Format" consist of two Files :
The Header-File (<Picture>.svo) and the Data-File (<Picture>).
The Header-File contains information about the Raw-Data in the
Data-File. For loading and saving the Graphics it makes no difference,
which FileName - Header-File or Data-File - is given.
This is managed by the svobject.

Because recognition of the Types of packed Files is only possible
by reading the whole File into a buffer and then check it, we use
two different Files for both : Graphics Information and Graphics.
The Data-File CAN be packed, the Header-File MUST NOT be packed.

The SVO.object does NOT save packed data by default, but you may either
specify an appropriate Xpk-Packer in the ControlPad or just pack it
with a Xpk-Packer-Program after writing.

The Data saved into the <Picture> File is read from a SV_GfxBuffer
structure, which has either been supplied by the calling application
or has been created from a supplied Screen-Pointer.
This data may be of type BITPLANE (like in an IFF-ACBM : BitPlane
after BitPlane, but without padding) or ONEPLANE (8 Bit ChunkyPixel
or 24 Bit RGB-Data).

This allows efficient packing and unpacking with PowerPacker and all
other XPK-Packers.

Remember, that the original Data written by the SVO.object may not
necessarily be packed and therefore might need a lot of diskspace.
So better pack it immediately after it has been written.

Source-Code
~~~~~~~~~~~
The "SVO Graphics File Format" is an "Open File Format", which means,
that everyone is invented to use and support it and to make suggestions
to improve the File Format or the Algorithms to handle it.

To allow support of the File Format and to give an example, how to write
your own superview.library compatible "svobjects", this distribution
contains the full SourceCode of the "SVO.svobject".

Feel free to use it for your own, commercial or non-commercial, programs !

History
~~~~~~~
```

V2.8 (01.07.1994) :

- writing of SVO-Files did not work since V2.6
- now it may be specified in "ENV:superview-Library/SVO.controlpad", whether written SVO-Files should be packed and which packer should be used ("PACKMETHOD=xxxx")
- SVO-Files now are written in the way the data has been stored in the GfxBuffer (either ONEPLANE or BITPLANE).
  Screens are transferred to BITPLANE data.
  Reading of such files was already possible, but not yet writing.
  This will sometimes allow more efficient packing with XPK
  (e.g. when ONEPLANE-8 data with only 16 Colors is used as input).

```
**********************************************************************
* Revison history entries before V2.8 have been deleted for space reasons *
**********************************************************************
```

1.20 GIF.svobject

© 1994-95 by Andreas R. Kleinert.
(Also see notes under "Credits".)
FREEWARE. All rights reserved.

Version : 2.12
Release Date : 22.07.1994

Description
~~~~~~~~~~~
GIF.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and needs any SVDriver with ChunkyPixel-Support.

It supports reading and writing of GIF Files.
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

A special 68030+ Version of GIF.svobject is called "GIF.svobject.030" and has to be copied/renamed to GIF.svobject before usage (it runs on the 68020 as well, but might be more suitable for the 68030 : Only the Compiler knows about it ... ;).
~~~~~~~~~~~

```
GIF.svobject works fine with Instruction Cache (Burst) and Data Cache
(Burst).


Bugs and other Problems
~~~~~~~~~~~~~~~~~~~~~~~
Fixed : The Decoding Routines are fully re-entrant since V2.2 !

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 !

Nevertheless for example it is possible to decode a picture at the same
time an other is been encoded, due to the fact, that the decoding
routines do not share any data or variables with the encoding routines.

History :
~~~~~~~~~
V2.12 (22.07.1994) :

 - fixed bad bug, which may have occured when writing GIFs :
   one sub-routine of the SVO_Write() function is not fully re-entrant,
   that’s nothing new (it is protected against being called twice
   at a time, anyway).
   The actual bug was, that for the second, third ... n-th
   call not all of the global static variables had been re-initialized
   again. This may have caused badly written GIFs with a high
   likelihood. Nevertheless there was a small chance to write
   more than one correct GIF.
   Anyway : This is now fixed !!
 - fixed small bug, which yet only occured once :
   GIFs may contain various "extension" blocks, which do not really
   have a defined length. This "dynamic length" feature was only
   correctly implemented for "comment" blocks, so that e.g. a big-sized
   "application" block would have caused the program to be kept in
   a never-ending loop (with a high likelihood).
   This small bug is now fixed.
   Maybe you’ve never been confronted with it.

***************************************************************************
* Revison history entries before V2.12 have been deleted for space reasons *
***************************************************************************
```

1.21 BMP.svobject

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 2.12
Release Date : 29.01.1995
```

Description
~~~~~~~~~~~

BMP.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and
needs any SVDriver with ChunkyPixel-Support.

It supports reading and writing of W*nd*ws V3.x (TM) BMP Files.
In detail these are :

Reading :

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

Writing :

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

Source Colors	Version	Type	Destination Colors
2	BMP V3.00	packed chk. pix.	2
4 .. 16	BMP V3.00	packed chk. pix.	16
32 .. 256	BMP V3.00	chunky pixel	256
(24 Bit)	BMP V3.00	RGB pixel	(24 Bit)

History
~~~~~~~

V2.12 (29.01.1995) :

- did not set correct (8 Bit) ViewMode for 24 Bit graphics (0xFFFFFFFF).
  Caused problems with some SVOperator-SVDriver calls, if not changed.
- speed increase in internal "top-to-bottom" routine

V2.11 (11.01.1995) :

- Oops: fixed 24 Bit reading/writing: what we used, were RGB-Pixels,
  but BMP uses BGR-Pixels. Fixed.

V2.10 (16.11.1994) :

- fixed bug, where the buffer of a written graphics would have been
  turned upsidedown after the operation (if source wasn’t a BitMap)

V2.9 (02.10.1994) :

- BMP.svobject was not able to make ScreenShots, because a source
  SV_GfxBuffer had always been required (and accessed: Enforcer-Hit ?)
  for some flag checks (instead of first generating the buffer from
  the Screen and then checking it).
  This has been fixed, ScreenSaving is possible now.

```
***************************************************************************
* Revison history entries before V2.9 have been deleted for space reasons *
***************************************************************************
```

1.22 WinIcon.svobject

© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

```
Version      : 2.1
Release Date : 06.11.1994
```

Description
~~~~~~~~~~~
WinIcon.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and
needs any SVDriver with ChunkyPixel-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
~~~~~~~
V2.1 (06.11.1994) :

 - first version.

1.23 FBM.svobject

© 1994-95 by Andreas R. Kleinert.
(Also see notes under "Credits".)
FREEWARE. All rights reserved.

```
Version      : 2.8
Release Date : 25.12.1994
```

Description
~~~~~~~~~~~
FBM.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and
needs any SVDriver with ChunkyPixel-Support.

It supports reading and writing of FBM (*nix) Files.
In detail these are :

Reading :

FBM 8 Bit, 1 Plane	FBM 8 Bit, 3 Planes
2..256 Colors; 256 Greyscales	24 Bit
~~~~~~~~~~~

```
  Writing :

   FBM 8 Bit, 1 Plane                  FBM 8 Bit, 3 Planes
   2..256 Colors                       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
  ~~~~~~~
  V2.8 (25.12.1994) :

   - added 24 Bit support (reading and writing)
   - fixed bugs in reading and writing of FBM files, which might
     have caused the "bits" value of the header to be interpreted
     wrong (especially when writing). Fixed now (must be "8" always).
   - colormapped 8 Bit files with less than 256 Colors will now be
     identified by the length of the colormap, so that e.g. a
     16 Color file will no longer be handled like a 256 Color file.

***************************************************************************
* Revison history entries before V2.8 have been deleted for space reasons *
***************************************************************************
```

1.24 PNM.svobject

```
  © 1994-95 by Andreas R. Kleinert.
  FREEWARE. All rights reserved.

  Version      : 2.1
  Release Date : 25.12.1994

  Description
  ~~~~~~~~~~~
  PNM.svobject is an external Library-Module for the superview.library V2+.

  It contains SVDriver-Support for the superview.library V3+, and
  needs any SVDriver with ChunkyPixel-Support.

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

  Reading :

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

   PBM (P4)         PGM (P5)          PPM (P6)
   Black & White    256 GreyScales    24 Bit TrueColor

  Writing :

   All input data, no matter how many Colors actually are used,
   will be written as binary 24 Bit PPM data (P6).
```

```
History
~~~~~~~
V2.1 (25.12.1994) :

 - first version
```

1.25 C64.svobject

```
© 1994-95 by Andreas R. Kleinert.
(Also see notes under "Credits".)
FREEWARE. All rights reserved.

Version      : 2.5
Release Date : 07.05.1994

Description
~~~~~~~~~~~
C64.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and
needs any SVDriver with ChunkyPixel-Support.

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

Reading :
```

Format	Dimensions	Colors	Displayed as	FileSize	Flexibility
Doodle	320x200	2/16	320x200	9218	(9200 < fs < 9230)
Koala	160x200	4/16	320x200	10003	(fs == 10003)

```
Writing :

 (not supported yet)

History
~~~~~~~
V2.5 (07.05.1994) :

 - use of SAS/C V6.51
 - use of new "superviewsupport.library" V1
   This saves some space again (here : ca. 1100 Bytes).

***************************************************************************
* Revison history entries before V2.5 have been deleted for space reasons *
***************************************************************************
```

1.26 IMG.svobject

```
  © 1994-95 by Andreas R. Kleinert.
  FREEWARE. All rights reserved.

  Version      : 2.7
  Release Date : 15.05.1994

  Description
  ~~~~~~~~~~~
  IMG.svobject is an external Library-Module for the superview.library V2+.

  It contains SVDriver-Support for the superview.library V3+, and
  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 Colors)
   - IMG (18 Byte Header) GreyScaled (1..8 Bits = 2..256 Colors)

  Writing :

   (not supported yet)

  History
  ~~~~~~~
  V2.7 (15.05.1994) :

   - fixed version history (V2.5 mentioned twice)
   - fixed subtype strings (one blank too much)

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

1.27 TIFF.svobject

```
  © 1994-95 by Andreas R. Kleinert.
  (Also see notes under "Credits".)
  FREEWARE. All rights reserved.

  Version      : 2.9
  Release Date : 13.01.1995

  Description
  ~~~~~~~~~~~
  TIFF.svobject is an external Library-Module for the superview.library V2+.

  It contains SVDriver-Support for the superview.library V3+, and
  needs any SVDriver with Bitplane-/ChunkyPixel-Support.

  It supports reading and writing of TIFF-Files (V5.0).
  In detail these are :
```

Reading :

Compression	Code	PixelBits	Planes	Max. Depth	As	Test
None	1	1	(system)	1*(system)	BITPLANE	
None	1	8	1,3	8 / 24	ONEPLANE	X
Mac PackBits	32773	1	(system)	1*(system)	BITPLANE	X
Mac PackBits	32773	4	1	4	ONEPLANE	X
Mac PackBits	32773	8	1	8	ONEPLANE	
LZW	5	8	1,3	8 / 24	ONEPLANE	(X)

Not all of these modes may work correctly and/or have been tested for working correctly. Tested were :

```
  - None        , PixelBits 8x1, ONEPLANE   (problems with odd-width pics)
  - None        , PixelBits 8x3, ONEPLANE   (problems with odd-width pics)
  - Mac PackBits, PixelBits 1x?, BITPLANE   (nearly no problems recognized)
  - Mac PackBits, PixelBits 4x1, ONEPLANE   (no problems recognized)
```

Writing :

Writing is only supported for Graphics upto 256 Colors.
They are always written as : Motorola (MM), None, 8, 1, ONEPLANE

Remarks

This SVObject was quite a hard piece of work and is still perhaps not fully free of bugs yet.

A Word About ...

... LZW compressed pictures :
sometimes pictures are saved very strangely, so that decoding with common code does not work at all. I’m sorry.

Supported

The following should work almost always :

```
 - reading of Intel- (II) and Motorola- (MM) Files
 - reading of 256 Color (8 Bit) Files,
   either packed with : - None
                        - Mac PackBits
 - reading of uncompressed 24 Bit files (RGB 3x8 Bits)
```

Not supported

Files with unknown compression types (e.g. CCITT Fax) are as well rejected as files with unsupported Compression/PixelBits/Plane combinations, like e.g. None/4/2.

Requests and Suggestions

If you have some FREELY DISTRIBUTABLE TIFF-Files, which do not work with this program, you may send them to me, so that I can try - at least _try_ - to improve this program to also allow reading of these.

```
  Debugging
  ---------
  Nevertheless usually information on a specific picture can be requested,
  although the picture itself cannot be displayed.
  Not all applications may support this (e.g. SuperView does not), because
  the error return value may prevent them against doing this.

  History
  ~~~~~~~
  V2.9 (13.1.95) :

   - detected some bugs/problems with multi-strip LZW files (fixed now),
     als found out, that several might not be read correctly at all
   - greatly speed-up on LZW code, re-wrote it in most of its parts
   - completely removed any of the CCITT-related code (no more try!)
   - updated docs and more


  V2.8 (18.12.94) :

   - V2.7 (at least the one, which came with V9.10) always popped
     up some requesters with debugging information while/after
     loading a TIFF file. Useful information, but unnecessary to
     be display _always_.
     Removed.

***************************************************************************
* Revison history entries before V2.8 have been deleted for space reasons *
***************************************************************************
```

1.28 EPS.svobject

```
  © 1994-95 by Andreas R. Kleinert.
  FREEWARE. All rights reserved.

  Version      : 2.1
  Release Date : 30.12.1994

  Description
  ~~~~~~~~~~~
  EPS.svobject is an external Library-Module for the superview.library V2+.

  It contains SVDriver-Support for the superview.library V3+, and
  needs any SVDriver with Bitplane-/ChunkyPixel-Support.

  It supports reading of the Trailers/Previews/Thumbnails which may be
  part of EPS.graphics. This will usually be TIFF graphics, but it
  makes not 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" to get informed, which TIFF
    file variations are currently supported).

  Reading :
```

(Any contained Trailers/Previews/Thumbnails, as long as the specific contained file format - if any - is supported by superview.library).

History
~~~~~~~
V2.1 (30.12.1994) :

- first version

1.29 Targa.svobject

Version   : 2.5
Release Date : 29.01.1995

Description
~~~~~~~~~~~
Targa.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and needs any SVDriver with Bitplane-/ChunkyPixel-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
(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
~~~~~~~
- Currently files with Colorbits other than 1, 8 or 24 (16, 32) are not supported yet.
- ColorMaps have to be of type "3-Byte RGB".
- The flags for "mirroring" Images vertically and/or horizontally are not fully interpreted yet, nevertheless they are checked and reported via 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
  ~~~~~~~
  V2.5 (29.01.1995) :

   - added support for RLE-encoded files (tested with 24 Bit graphics)
   - speed increase in internal "top-to-bottom" routine

  V2.4 (26.06.1994) :

   - fixed small bug, which might have caused this one crashing/rejecting,
     when writing from a Screen (instead of a GfxBuffer).

***************************************************************************
* Revison history entries before V2.4 have been deleted for space reasons *
***************************************************************************
```

1.30 WPG.svobject

```
  © 1994-95 by Andreas R. Kleinert.
  FREEWARE. All rights reserved.

  Version      : 2.4
  Release Date : 22.05.1994

  Description
  ~~~~~~~~~~~
  WPG.svobject is an external Library-Module for the superview.library V2+.

  It contains SVDriver-Support for the superview.library V3+, and
  needs any SVDriver with Bitplane-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

  Writing :

   (not supported yet)

  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, this will
  be displayed, 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 FileInfoRequester.

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 should not get any problems with those pictures.

History
~~~~~~~
V2.4 (22.05.1994) :

- fixed bug, which caused the last 4 Bytes of the ColorMap not to be read (last two Colors have been wrong, then).
  This did not concern grey-scaled Files without own ColorMaps ;-)
- fixed the same bug for BitMap data : the last 10 Bytes might not have been read.
- modified Doc-File slightly to reflect, that 8 Bit-Files are read correctly (not just since V2.4 ...)

```
*************************************************************************
* Revison history entries before V2.4 have been deleted for space reasons *
*************************************************************************
```

1.31 SunRaster.svobject

© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

```
Version      : 2.2
Release Date : 24.12.1994
```

Description
~~~~~~~~~~~
SunRaster.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and needs any SVDriver with Bitplane-/ChunkyPixel-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
~~~~~~~
 - Currently files with Colorbits other than 1, 8 or 24
   are not supported yet.
 - only RGB-plane ColorMaps supported (or monochrome, without map)
 - max. 1 plane of BitMap data allowed

History
~~~~~~~
V2.2 (24.12.1994) :

 - when writing 24 Bit files, Colormap specification would have been
   wrong, which e.g. would have confused NetPBM.
   Also the wrong planetype flag had been set. Fixed.

V2.1 (26.06.1994) :

 - first version
```

1.32 Pictor.svobject

```
Version      : 2.1
Release Date : 13.06.1994

Description
~~~~~~~~~~~
Pictor.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and
needs any SVDriver with Bitplane-/ChunkyPixel-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).

Writing :

 (not yet supported)

History
~~~~~~~
V2.1 (13.06.1994) :
```

```
  - first version
```

1.33 MAC.svobject

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 2.1
Release Date : 12.05.1994

Description
~~~~~~~~~~~
MAC.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and
needs any SVDriver with Bitplane-Support.

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

Reading :

 - MAC Black & White 576x720

Writing :

 (not supported yet)

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. Due to the fact, that we can't
just check the suffix or prefix of the file to be "MAC", we have to
check the filetype also.
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 !

History
~~~~~~~
V2.1 (12.05.1994) :

 - first version
```

1.34 JPEG.svobject

```
This one is a "bonus product", which has been written
by me and has been distributed separately upto version 8.2.
```

You find it - together with its documentation and an appropriate Installer-Script - inside the Directory "Bonus".

Show Documentation

1.35 PCD.svobject

This one is a "bonus product", which has been written by me and is not meant as an integral part of the main archive.

You find it - together with its documentation and an appropriate Installer-Script - inside the Directory "Bonus".

Show Documentation

1.36 YUVN.svobject

© 1994-95 by Andreas R. Kleinert.
(Also see notes under "Credits".)
FREEWARE. All rights reserved.

```
Version      : 2.3
Release Date : 02.01.1995
```

Description
~~~~~~~~~~~

YUVN.svobject is an external Library-Module for the superview.library V2+.

It contains SVDriver-Support for the superview.library V3+, and needs any SVDriver with 24 Bit ChunkyPixel-Support.

It supports reading and writing of IFF-YUVN (YUVN) Files. In detail these are :

Reading :

- IFF-YUVN 24 Bit (Y, 1/4 U, 1/4 V)

Writing :

- IFF-YUVN 24 Bit (Y, 1/4 U, 1/4 V)

Reading/writing from/to ClipBoard is supported.

Remarks
~~~~~~~

- only 24 Bit sources will be written as IFF-YUVN. There’s no conversion done from e.g. 256 Colors to 24 Bit to perform the requirements of this file format.

```
Idea
~~~~
 The integration of this file format has been suggested by Detlef Winkler,
 who asked me to do some more support for amiga-like 24 Bit file formats,
 as e.g. IFF-24.
 He sent me some digitized IFF-YUVN graphics, so that I was able to
 analyze the file format and to add support to superview-Library.
 Thanks !

History
~~~~~~~
V2.3 (02.01.1994) :

 - fixed the same bug as in V2.2, now also for encoding/writing

V2.2 (29.12.1994) :

 - Y-values (brightness) had not been interpreted in the right way

V2.1 (15.10.1994) :

 - first version with both, read- and write support
   (reading/writing from/to ClipBoard also is possible)
```

1.37 ECS.svdriver

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 1.12
Release Date : 29.01.1995

Requirements
~~~~~~~~~~~~
- OS V2.04+ (V37)+ and its libraries
- superviewsupport.library V1+
- AMIGA with Enhanced Chip Set (ECS)

Description
~~~~~~~~~~~
ECS.svoperator is an external SV_GfxBuffer display module
for superview.library V3+.

It supplies functions for V2+ SVObjects to allow object-orientated
displaying of graphics on ECS Screens.

ECS.svdriver is identical to AGA.svdriver, except to the "chunky pixel"
support and other features, which have been realized without V39 functions
here.

This Driver supports the following :

Dimensions        Depth         Type

[ECS]             [ECS]         BITPLANE
```

```
 [ECS]            8            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.

 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=ExtractGrayScales
                        ; e.g. "24BITOPERATOR=24BitToHAM"
                        ; or   "24BITOPERATOR=ExtractGrayScales"
                        ; (case-sensitive, ".svoperator" may be added)
                        ; specifies, which operation should be performed
                        ; on 24 Bit graphics _before_ displaying them
                        ; (if not specified, "on error" default handling
                        ;  takes place)

 Known Bugs
 ~~~~~~~~~~
 There's a bug in the chunky pixel routines of OS V37-40, which has been
 worked-around in a way, that the last (upto) 7 pixels of a chunky
 graphics, which has a width ending not on a byte boundary (as e.g. 433)
 will be cut off.
 That's not really a good solution, but better than crashing the whole
 system.
 Maybe this bug will be fixed in future OS or chipset versions (maybe
 this problem would not happen with the CD-32's Chunky Pixel Hardware ?!).

 History
 ~~~~~~~
 V1.12 (29.01.1995) :

  - added SVOperator-Support for 24 Bit graphics:
    via specifying "24BITOPERATOR" within "ECS.controlpad" you may
    decide which operation should be performed on 24 Bit graphics
    _before_ displaying them (default rejecting them)

 V1.11 (28.12.1994) :

  - fixed bug in (re-)initialization module

 V1.10 (28.09.1994) :

  - WritePixelLine8() sometimes seems to destroy the source buffer.
    Fixed now by using a copy of the corresponding line content.

 V1.9 (05.09.1994) :

  - added work-around for chunky-pixel bug in the OS
    (see library-docs for V8.2)

***************************************************************************
* Revison history entries before V1.8 have been deleted for space reasons *
```

****

1.38 AGA.svdriver

```
Version      : 1.15
Release Date : 28.12.1994
```

Requirements
~~~~~~~~~~~~~

- OS V3.00+ (V39)+ and its libraries
- 68020+
- superviewsupport.library V1+
- AMIGA with AGA ChipSet (ECS still supported, but with restrictions) or an appropriate Graphics Card with Workbench Emulation

Description
~~~~~~~~~~~~

AGA.svoperator is an external SV_GfxBuffer display module
for superview.library V3+.

It supplies functions for V2+ SVObjects to allow object-orientated
displaying of graphics on AGA Screens.

AGA.svdriver is identical to ECS.svdriver, except to the "chunky pixel"
support and other features, which have been realized with V39 functions
here.

This Driver supports the following :

Dimensions	Depth	Type	
[AGA]	[AGA]	BITPLANE	
[AGA]	8	ONEPLANE	(Chunky Pixel)

ONEPLANE 24 Bit Graphics (RGB 3x8-ONEPLANE) will be displayed as
"best fitting of 256 Colors" by default.

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=ExtractGrayScales
                        ; e.g. "24BITOPERATOR=24BitToHAM"
                        ; or   "24BITOPERATOR=ExtractGrayScales"
                        ; (case-sensitive, ".svoperator" may be added)
                        ; specifies, which operation should be performed
                        ; on 24 Bit graphics _before_ displaying them
                        ; (if not specified, "on error" default handling
                        ;  takes place)
```
~~~~~~~~~~~~~~~~~~~

Known Bugs

There’s a bug in the chunky pixel routines of OS V37-40, which has been worked-around in a way, that the last (upto) 7 pixels of a chunky graphics, which has a width ending not on a byte boundary (as e.g. 433) will be cut off.
That’s not really a good solution, but better than crashing the whole system.
Maybe this bug will be fixed in future OS or chipset versions (maybe this problem would not happen with the CD-32’s Chunky Pixel Hardware ?!).

History

V1.15 (28.12.1994) :

- fixed bug in (re-)initialization module

V1.14 (10.12.1994) :

- added SVOperator-Support for 24 Bit graphics:
  via specifying "24BITOPERATOR" within "AGA.controlpad" you may decide which operation should be performed on 24 Bit graphics _before_ displaying them (default is "picking best of 256 Colors")

V1.13 (05.11.1994) :

- when running under OS 3.1 (V40) we now do use WriteChunkyPixels() instead of WritePixelLine8().
  Because the specific bug in these routines (still) has not been fixed, we cannot write the whole array (or even full line, when odd width), but it may be faster nevertheless (no TmpRas, no separate Bitmap), at least on a CD-32.
  This will work in 8 and 24 Bit (dither) mode.

V1.12 (21.10.1994) :

- V1.11 had a small (?) bug: autoscrolling of large pictures had been disabled while fixing several other things (forget to re-activate) it. Sorry.
  Now it will work again (BTW: ECS.svdriver did not have this bug).

V1.11 (28.09.1994) :

- WritePixelLine8() sometimes seems to destroy the source buffer.
  Fixed now by using a copy of the corresponding line content.
- closed "memory black hole" in 24 Bit-support :
  there was always a memory loss of <width> bytes, when displaying a 24 Bit graphics (linebuffer not freed)

V1.10 (05.09.1994) :

- added work-around for chunky-pixel bug in the OS
  (see library-docs for V8.2)

```
*****************************************************************************
* Revison history entries before V1.10 have been deleted for space reasons *
```

```
****************************************************************************
```

1.39 EGS.svdriver

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 1.8
Release Date : 28.12.1994

Requirements
~~~~~~~~~~~~
- OS V2.04+ (V37)+ and its libraries
- superviewsupport.library V3+
- egs.library      V1+
- egsintui.library V1+
- egsgfx.library   V1+
- AMIGA with EGS Graphic Card (or EGS distribution with Amiga Emulation)

Description
~~~~~~~~~~~
EGS.svoperator is an external SV_GfxBuffer display module
for superview.library V3+.

It supplies functions for V2+ SVObjects to allow object-orientated
displaying of graphics on EGS Screens (Enhanced-Graphic-System Screens).
All corresponding EGS-Libraries are needed.

This Driver supports the following :

Emulation   Dimensions      max. Depth    Type

EGS-Card    [EGS-Card]      8/24          ONEPLANE
                            8             BITPLANE
ECS/AGA     [ECS/AGA]       8             ONEPLANE
                            8             BITPLANE

Currently BITPLANE with more than 8 Bit Colordepth and ONEPLANE with other
pixelbits than 8 or 24 (e.g. 16 Bit R5:G5:B5:1) are not supported.

EGS-Cards
---------
When using an EGS Card, all supported ColorDepths should be displayed
correctly in TrueColor.

Amiga ECS/AGA emulation
-----------------------
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 ...

Closing of the current EGS Display is possible by freeing the handle's resources or delocating the handle, but no screen or window addresses (for IDCMP checking) are returned by the related library functions, because EGS is (as far as I know) binary incompatible to Intuition.
To fix this a little bit, every time when displaying a graphic on the EGS-Screen a Workbench Window is opened which allows usual IDCMP access.
Window IDCMP and Window Flags (except backdrop/borderless) are recognized and used.

History
~~~~~~~
V1.8 (28.12.1994) :

- fixed bug in (re-)initialization module

```
***************************************************************************
* Revison history entries before V1.8 have been deleted for space reasons *
***************************************************************************
```

1.40 OPAL.svdriver

This one is a "third party product", which has been written by Steve Quartly and Paul Huxham.

You find it - together with its documentation and an appropriate Installer-Script - inside the Directory "Bonus".

Show Documentation

1.41 XOR.svoperator

© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

```
Version      : 1.4
Release Date : 28.12.1994
```

Description
~~~~~~~~~~~
XOR.svoperator is an external SV_GfxBuffer modification module for superview.library V9+.

```
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
~~~~~~~
V1.4 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.3 (01.11.1994) :

 - now rejects 24 Bit data (would have produced garbage)

V1.2 (09.10.1994) :

 - there was a big memory black hole in V1.1 (internal memory lists
   had not been delocated). Fixed.

V1.1 (30.09.1994) :

 - first version
```

1.42 24BitToHAM.svoperator

```
© 1994-95 by Andreas R. Kleinert.
(Also see notes under "Credits".)
FREEWARE. All rights reserved.

Version      : 1.4
Release Date : 09.02.1995

Description
~~~~~~~~~~~
24BitToHAM.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
```

```
~~~~~~~
V1.4 (09.02.1995) :

 - "HAM6_WELL" procuded "HAM8_QUICK" and vice versa. Fixed.


V1.3 (05.01.1995) :

 - modified "24BitToHam6.svoperator" to create HAM8 output,
   if requested, also added "quick" options for both, HAM6 and HAM8
 - renamed to "24BitToHAM.svoperator"

V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (04.12.1994) :

 - first version
```

1.43 Dither24Bit.svoperator

```
© 1994-95 by Andreas R. Kleinert.
(Also see notes under "Credits".)
FREEWARE. All rights reserved.

Version      : 1.1
Release Date : 26.12.1994

Description
~~~~~~~~~~~
Dither24Bit.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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

Technically, a modificated version of Heckbert's median cut is applied
to Floyd-Steinberg dithered data.

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)

History
~~~~~~~
V1.1 (26.12.1994) :

 - first version
```

1.44 HilbertDither256.svoperator

```
© 1994-95 by Andreas R. Kleinert.
(Also see notes under "Credits".)
FREEWARE. All rights reserved.

Version      : 1.3
Release Date : 28.12.1994

Description
~~~~~~~~~~~
HilbertDither256.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
~~~~~~~
V1.3 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.2 (09.10.1994) :

 - added ControlPad-Switch for selection of the background color

V1.1 (09.10.1994) :

 - first version
```

1.45 24BitToHAM.svoperator

```
© 1994-95 by Andreas R. Kleinert.
(Also see notes under "Credits".)
FREEWARE. All rights reserved.

Version      : 1.4
Release Date : 09.02.1995
```

```
Description
~~~~~~~~~~~
24BitToHAM.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
~~~~~~~
V1.4 (09.02.1995) :

 - "HAM6_WELL" procuded "HAM8_QUICK" and vice versa. Fixed.


V1.3 (05.01.1995) :

 - modified "24BitToHam6.svoperator" to create HAM8 output,
   if requested, also added "quick" options for both, HAM6 and HAM8
 - renamed to "24BitToHAM.svoperator"

V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (04.12.1994) :

 - first version
```

1.46 AnyTo24Bit.svoperator

```
Version      : 1.1
Release Date : 09.02.1995

Description
~~~~~~~~~~~
AnyTo24Bit.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
~~~~~~~
```

```
V1.1 (09.02.1995) :

 - first version
```

1.47 ExtractGrayScales

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 1.3
Release Date : 29.01.1995

Description
~~~~~~~~~~~
ExtractGrayScales.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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 grayscaled
                        ; output (1->2 Colors .. 8->256 Colors)

History
~~~~~~~
V1.3 (29.01.1995) :

 - added ControlPad-Support for creating grayscaled graphics with
   less than input colors (greyscales); even 24 Bit may now be
   scaled to less than 256 grayscales

V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (30.09.1994) :

 - first version
```

1.48 ExtractRed

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.
```

```
Version      : 1.2
Release Date : 28.12.1994

Description
~~~~~~~~~~~
ExtractRed.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
~~~~~~~
V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (29.09.1994) :

 - first version
```

1.49 ExtractGreen

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 1.2
Release Date : 28.12.1994

Description
~~~~~~~~~~~
ExtractGreen.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
~~~~~~~
V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (29.09.1994) :

 - first version
```

1.50 ExtractBlue

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 1.2
Release Date : 28.12.1994

Description
~~~~~~~~~~~
ExtractBlue.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
~~~~~~~
V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (29.09.1994) :

 - first version
```

1.51 TopToBottom

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 1.2
Release Date : 28.12.1994

Description
~~~~~~~~~~~
TopToBottom.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
~~~~~~~
V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (01.11.1994) :

 - first version
```

1.52 LeftToRight

```
© 1994-95 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 1.2
Release Date : 28.12.1994

Description
~~~~~~~~~~~
LeftToRight.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
~~~~~~~
V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (01.11.1994) :

 - first version
```

1.53 Rotate

```
© 1995 by Andreas R. Kleinert.
FREEWARE. All rights reserved.

Version      : 1.1
Release Date : 02.01.1995

Description
~~~~~~~~~~~
Rotate.svoperator is an external SV_GfxBuffer modification module
for superview.library V9+.

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
~~~~~~~
```

```
  V1.1 (02.01.1995) :

   - first version
```

1.54 Scale50

```
  © 1995 by Andreas R. Kleinert.
  FREEWARE. All rights reserved.

  Version      : 1.1
  Release Date : 02.01.1995

  Description
  ~~~~~~~~~~~
  Scale50.svoperator is an external SV_GfxBuffer modification module
  for superview.library V9+.

  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
  ~~~~~~~
  V1.1 (02.01.1995) :

   - first version
```

1.55 Used literature for developing this program

```
Literature:
~~~~~~~~~~~
  [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] ... and perhaps books/magazines/articles, which I don't remember yet !
```

1.56 Credits

```
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.

  Also, the code has been rewritten to also be abled to create palette-
  based HAM8-output.
  The code used for the "quick" option of the SVOperator (without palette)
  is not related in any way to code of the FBM package.

  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
   *
   [...]


  Dither24Bit.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 to be strongly modfied to work with common
  24 Bit RGB Pixels instead of R-G-B planes and has been made
  fully re-entrant (also various other changes had to be done).
  Only the basic algorithm is similar to the original code.
  Here's the copyright notice as found in "fbquant.c"
  (revision headers cut off / left out) :

  /****************************************************************
   * fbquant.c: FBM Release 1.0 25-Feb-90 Michael Mauldin
```

```
 *
 * 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.
 [...]
```

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).

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 ’giftorle’, 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 GIFENCODE 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 GIFENCODE 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)
 *
```

TIFF.svobject

Some parts of this software are based in part on code from the "Aldus Developers Desk" Release 90-06-14 (as found on the Nova Media "Grafik-Collection I CDROM", 1993, Directory "ZIP", File "TIFFRD.ZIP").

Many work had to be done, to get that code working on the Commodore Amiga (or even compileable under SAS/C V6.51).
(Actually I’m not sure, if it really works ...)
In detail, I’m making use of strongly modified versions of the lzw (lzwde.c) and CCITT 1D (tiff2.c) decoding routines.

YUVN.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).
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.svperator
  --------------------------
  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" (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).
```

1.57 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.

  \textdegree{} For usage of the IFF-based SVObjects, like ILBM, ACBM and YUVN,
    you should take care to have iffparse.library V37+ on your bootdisk.
  \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.

Additionally there are some more requirements, which do not
necessarily depend on the OS or the Hardware:

  \textdegree{} superviewsupport.library V1+ upto V5+,
    where the version depends on the SVObjects/SVDrivers/SVOperators.
    This distribution always includes the latest version of the library.

Some SVObjects,SVDrivers or SVOperators additionally require
different Hardware/Software configurations, but usually this is
stated within their own documentation:

  \textdegree{} AGA.svdriver   OS 3.x+  68020+ with ECS, AGA or GraphicsCard
                           (not fully featured with ECS)
  \textdegree{} OPAL.svdriver           OpalVision Card
  \textdegree{} JPEG.svobject           68020+
  \textdegree{} SVO.svobject            xpkmaster.library V2+
                           which is (C)opyright by its authors
                           (c/o Urban Dominik Mueller)
  \textdegree{} GIF.svobject.030        68020+
                           (if you don’t own one, just use GIF.svobject)
```