SuperView-Library

COLLABORATORS

	TITLE : SuperView-Library		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		December 7, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

SuperView-Library

1.1   SuperView Library Documentation

superview.library V11.7

- FREEWARE -

© 1993-95 by Andreas R. Kleinert. All rights reserved.

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

Release Date: 27.5.1995

Copyrights
Disclaimer
Distribution
Requirements
How to install and configure it
Short: Purpose and Abilities
Supporting Software
SVObject Descriptions/History
SVDriver Descriptions/History
SVOperator Descriptions/History
Possible future enhancements ...
History
Thanks and Greetings
Used literature for developing
Credits
How to contact the author

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

Die CHAOS-Theorie:
"Dabei geht es zum Beispiel darum, den verdammten
Schmetterling zu finden, dessen Flügelschläge die

```
                        vielen Stürme in letzter Zeit verursacht haben."
                        (Terry Pratchett in "Total verhext")


                   Dedicated to all people, who actually READ docfiles ;-)
```

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.

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

Table of contents: \textdegree{} Non-commercial distribution and usage
                      \textdegree{} Commercial usage / distribution licenses
                      \textdegree{} "Agreement-by-usage" condition

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.
Also CD-ROMs by Fred Fish, AmiNet and ADX (Amiga Demos and Tools).

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 distri-
butable 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

```
Main Installation
~~~~~~~~~~~~~~~~~~
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/" !

Customized Installation for 680x0 systems
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
When needed, you may also install some of the special 68030(020)
and 68040 versions of some of the library modules.
These will not yet be automatically installed via the script,
so you should copy them to LIBS: just after the main installation
(just copy "libs/68030" or "libs/68040" to your LIBS:-directory).

Datatypes-Support for OS 3.x
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
"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 !!!

Add-On's
~~~~~~~~
The "Bonus" directory of the library archive does contain several
addon-modules which might wished to be installed.
This has to be done separately.

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

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

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
at a Ram-Disk, because most programs will perhaps only try to place
files there, after they already checked, that there's no more
memory available (so even your Ram-Disk would not grant more space).

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

Preferences
~~~~~~~~~~~
For modifying any Preferences you should take a look into the
file "ControlPads", where several optional settings are described.

For editing ControlPad Files you need a TextEditor, although
SuperViewPrefs may be used as a shell to manage this.

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 8/24bit Files
EPS	EXT	(various)	(not yet available)
Targa	EXT	max 8/24bit Files	max 8/24bit Files
WPG (W*rdP*rf*ct)	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)
FastILBM24	EXT	24bit Files	(-> via IFF-ILBM)
QRT/POV	EXT	24bit Files	24bit Files
Utah RLE	EXT	24bit Files	24bit Files

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

Planned SVObjects (no guarantee !) :

 - RIFF-BMP
 - Mac PICT
 - MTV
 - SGI
 - 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
CyberGraphics	CyberGraphics RTG System	8bit	8/24bit
EGS-System	EGS-Graphic-Libraries	8bit	8/24bit
OpalVision	OpalVision Card	8bit	8/24bit
Retina	Retina ZII/ZIII Card	8bit	8/24bit

 NOT INCLUDED (but e.g. on AmiNet)

| Picasso II | PicassoII-Card |
| Merlin | Merlin Graphics Card |

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

Planned SVDrivers (no guarantee !) :

 - PiccoloSD64/EGS7 and any other cards I get my hands on... ;-)

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
   Crop                "crops" boxes of any size from 8/24 Bit graphics
   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      THE program for use with superview.library
    - SimpleView     (superview.library Example Program)
    - MicroView      (superview.library Example Program)
    - KFracPlus      (fractal generator with superview-based saving option)
    - SIP            (for examinations)
    - DRAFU          (unreleased Beta-Version; ask for a demo)
    - SuperLoader    (Steve Quartly's OpalVision-Loader)
    - SqOpal         (by Steve Quartly; steveq@sndcrft.DIALix.oz.au)
    - ImageConvert   (by Andre Bergmann)
    - SVShow         (by Andre Bergmann)
    - Image Engineer (by Simon Edwards; s9407349@yallara.cs.rmit.edu.au)
    - ...

   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 only the InstallerScript-Icon is used yet
    (several times ;-)

 * 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
    and new Drawer-Icons.
    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 ;-) :

      Andre   Bergmann   made bug-reports and uses superview.library
      Alex   Carbin   TuC / Co-Sysop Century
      Rüdiger Dombrowski   ADX-Datentechnik, "Amiga Demos+Tools" CD-ROMs
      Sven   Drieling   Indy - Creator of "PowerBrei" DiskMag
      Simon   Edwards   author of Image Engineer
      Oliver   Eichhorn   author of EGSTasy
      Thomas   Eigentler   Programmer of Merlin-SVDriver (AmiNet)
      Thomas   Fischbach   Sysop of "GM-Box", my current Fido-Boss
      Fred   Fish   AmigaLibDisks and Fish CD-ROMs
      Stefan   Grad   GPD-Disks
      Klaus   Holtorf   for several information on graphic file formats
      Paul   Huxham   (Co-)Programmer of OpalVision-SVDriver
      Irseesoft   for supplying the PiccoloSD64 Graphics Cards
      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.
   Patrick  Ohly         Programmer of PicassoII-SVDriver (AmiNet)
   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 ... :-)
   Florian Zeiler        Sending me the Retina enabled me to write a Driver.

 - 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 :-)
   Jürgen       Schneider
   Klaus        Stengel
   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)

         When calling via phone you may leave a message,
         if I'm not available. - Andy

 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(/030/040/060)(T)/DraCo, Kickstart/OS Version,
 RAM, HardDisk, GfxCard, any special Hardware/Software).
Since their recently have been some problems: please don't forget
to mention, whether your machine is accelerated in any way, or
not (68020/.../060, 68881/882).

1.13   The future of the SuperView.library

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

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

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

I'll try to do this, but I hope that there's enough support from the
Amiga programmers all over the world (via Inter/AmiNet, Fido, etc.)
to help me to continue doing so.
This does not only mean fincancial support, but also additional
programming efforts like third-party improvements on the library
(why not write your own SVObjects/SVDrivers/SVOperators ?)

- Andy

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
Version 11	Version 1,2	Version 1,2	Version 1

   V11.7 (27.5.1995) :

```
            ------------------
            - New features:
              - included "CyberGraphics.svdriver" for CyberVision and
                any other GfxCards, which are supported by CyberGraphics.
                There have already been single releases of this SVDriver.
                (thanks to Ingenieurbüro Helrich for the PiccoloSD64 Card)
            - Fixes:
              - Retina.svdriver did not fail on missing retina*.libraries,
                but instead when trying to display anything (Guru). Fixed.
              - fixed bug within AGA.svdriver
            - Misc:
              - added some new Icons, e.g. SuperViewPrefs.info now contains
                the "selected" part of the ConfigFile-Icon of SuperView (nice)
                and the Drawer Icons have been replaced by a new one from
                Detlef Winkler
              - revised SuperViewPrefs to allow also settings for SVOperators,
                also completely revised GUI and soem more things.
            - Documentation and Programmers Section:
              - again reworked some parts of the documentation
              - added note on availability of Merlin.svdriver V1.1
            - Installation:
              - completely rewrote Installer-Script; e.g. libraries may
                now be installed to an other directory than LIBS:
                (for those who make use of "Assign LIBS: [PATH] ADD")
              - added configuration for CyberGraphics
              - ECS and AGA settings now also included with some specific
                GfxCard configurations (BITMAPCOPY=RTG)


        V11.6 (17.5.1995) :
        ------------------
        - Fixes:
          - superviewsupport.library's new C2P routines had been buggy
            for "less than 256 color graphics"
          - small fix on Retina.svdriver
        - Installation:
          - added sample configuration for RetinaZ2/Z3, OpalVision
            and so on (Docs/SampleConfigs)
          - Installer-Script had a bug within the SampleConfig Installation
            part. Also now asking sooner for basic system configuration.


        V11.5 (14.5.1995) :
        ------------------
        - added example program, which tests SVL_GetFileType() and
          displays some (useful ?) information
          (-> Steve Quartly)
        - Fixes:
          - revised "chunky to planar" conversion routines of
            superviewsupport-library
          - fixed _strange_ bug in writing ILBMs
          - ILBM, ACBM now do handle "foreign" CAMG ViewModes better
            (-> Jürgen Schäfer)
          - again fixed AnyTo24Bit concerning HAM ViewModes
            (-> Jürgen Schäfer)
        - 680x0 support:
          - added 68030 and 68040 version of superviewsupport.library
            (due to new default c2p/p2c conversion routines)
          - added 68030 and 68040 version of ILBM.svobject
```

```
        - added sample configuration for GfxCards

    V11.4 (7.5.1995) :
    ------------------
    - Fixes:
      - ECS.svdriver sometimes tried to open 24 Bit Screens, which
        seemed to cause serious crashes sometimes (-> Jürgen Schäfer)
      - 24BitToHAM contained some garbage; revised memory management
        dito for AnyTo24Bit
        dito for Dither24Bit
        dito and more for GIF
      - modified Dither24Bit to usually set color 0 as black
    - Documentation:
      - revised and updated

    V11.3 (1.5.1995) :
    ------------------
    - General:
      - from now on using SAS/C V6.55 for any new compilations
        (recompiled anything so far, with different results ;-)
      - now there are "libs" subdirs for 68030(020) and 68040 versions
    - New features:
      - added "Retina.svdriver" for Retina Z2/Z3 Cards and
        their 24 Bit Workbench Emulation System
        (thanks to Florian Zeiler/Irseesoft for the card)
      - ECS/AGA.svdrivers now have a new switch (BITMAPCOPY=RTG),
        which enables them to work with more Workbench Emulations
        than before when displaying BITPLANE type GfxBuffers
      - added new "QRT.svobject" for reading and writing of POV Ray's
        QRT Dump files (24 Bit)
      - introduced new "CPInfo" File Format, which describes
        possible ControlPad entries for any modules
        (-> Simon Edwards)
      - added "ENV"-Directory (containing .cpinfo-files) to the archive,
        which will be copied during the installation procedure
      - added new versions of JPEG, PCD and Unpack (see there)
      - added new "Crop.svoperator" for extracting parts of given
        GfxBuffers (specifying LeftEdge, TopEdge, Width and Height)
        (-> Steve Quartly, Jürgen Schäfer)
    - 680x0 support:
      - created "libs/68030" and "libs/68040" dirs for those binaries
        (to be copied by hand from there to LIBS:)
      - added 68040 version of GIF.svobject
        ("libs/68040/svobjects")
      - added 68030(020) and 68040 versions of TIFF.svobject
        ("libs/68030/svobjects" and "libs/68040/svobjects")
      - added 68030(020) and 68040 versions of PCD.svobject
        ("Bonus/svoPCD/libs/68030/svobjects" and ".../68040/svobjects")
    - Reworks:
      - any of the supplied libraries (superview, superviewsupport)
        now save A1 when opened/closed (anyway ;-)
        (only Opal.svdriver does not yet; don't know what about Picasso)
    - Fixes:
      - ANOTHER BIG BUG !
        All SVOperators did not load under 2.x, since V39 of some
        libs had been requested. Fixed.
        (Also updated svsupport requirements to V5)
```

```
        - FastILBM24.svobject now returns HAM8 with Hires ViewMode
        - AnyTo24Bit.svoperator did not recognize HAM, if ViewMode
          had been changed before (-> Jürgen Schäfer)
        - 24BitToHAM.svoperator did not initialize NativeDIPF field
        - some SVOperators may just have copied the Version and NativeDIPF
          field without verifying the size of the GfxBuffer structure
          or actually copying the whole structure. Fixed by just setting
          the actually supported (needed) version number.
        - fixed bug within Targa.svobject, which might have confused
          some HAM6/8 detection routines
      - Documentation and Programmers Section:
        - again revised and updated several docfiles
        - fixed some "bugs"
        - added note on UtahRLE (Raytracer) support, which is available
          separately (e.g. on AmiNet)
        - added note on availability of Picasso.svdriver V2.0


      V11.2 (1.4.1995) :
      ------------------
      * There have been several beta and one prerelease versions of V11.1
      * This is the final release. Discard the others.
      * For reasons of clearness I called it V11.2 now.

      - New features:
        - enhanced SVDriver specification (V2) to allow GfxBuffer-sensitive
          ScreenModeList generation, which might be used by external
          applications (e.g. SuperView).
          (-> Patrick Ohly)
          Also added new function to the library which makes use of this
          feature.
        - enhanced SV_GfxBuffer specification (V2) to allow preserving
          of the special information on the original screenmodes of
          graphics. This greatly simplifies handling of HAM, EHB and
          other special data. Uses standard "DIPF_" defines to prevent
          us from creating more dumb standalone-definitions.
          (-> Patrick Ohly)
        - overall re-worked ECS, AGA and EGS.svdriver and upgraded them
          to V2 specifications.
        - added HAM6/EHB colormap check to ECS/AGA SVDriver (64->32/16),
          also rewrote parts of the ViewMode generation/validation stuff
        - ILBM, ACBM and SVO now fill in NativeDIPF field of V2 GfxBuffers
        - changed BadCAMG recognition of ILBM and ACBM,
          also fixed a bug within that code (relict of OS 1.3)
        - added new "FastILBM24.svobject" for fast, immediate and low-memory
          loading, dithering and displaying of 24 Bit ILBM graphics
        - TIFF now also writes 24 Bit data.
      - Bug-Fixes:
        - BIG BUG!
          All SVOperators had been compiled in 68020 code, thus might
          have caused 68000 systems to crash. Fixed.
        - english "superviewlib.ct" translation file contained "deutsch"
          as language name. Did not matter ('cause dummy file), but fixed
          anyway
        - fixed small bug within SVL_FreeResources()
          (obviously never occured; there was a semicolon directly
           after an "check for NULL" if-construct)
        - fixed bug, which might have occured, when multiply displaying
```

```
        on different SVDrivers with the same handle (Open, Close,
        ChangeDisp, Open, Close, ...)
      - "Scale50.svoperator" sometimes failed on downscaling odd-width/
        height graphics
        (-> Jürgen Schäfer)
      - "PCD.svobject" now works 100% correctly (no longer blue-toned)
      - "TIFF.svobject" again debugged itself always (requester) in V10.2
      - included new JPEG.svobject V2.6 (use "AVAILMEM=<Bytes>" controlpad,
        if it crashes on your system; values >= 1000000 if possible)
    - Misc:
      - "libs" directory of the library archive contained PCD.svobject,
        although it should only be available in "bonus" dir.
        This caused some beta versions to also contain an obsolete version.
      - Scale50, Rotate, ExtractRed, ExtractGreen, ExtractBlue, Extract-
        GrayScales and LeftToRight operators now reject HAM
      - Amiga-foreign gfx formats (thus missing ViewMode field) with
        8 Bit saving option now also reject HAM as source for writing
    - Documentation and Programmers Section:
      - updated Docs, included info on Patrick Ohly's PICASSO-SVDriver
      - the header of this docfile now contains a nice statement
        (in german ;-)
      - added "SupportBBS" text !
      - updated Crashes.doc :-(


    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 inter-
        actively (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)

  ***************************************************************
  * Revison history entries below V10.1 have been deleted     *
  *                     to save diskspace !                   *
  ***************************************************************
```

1.15 ILBM.svobject

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

Version   : 2.20
Release Date : 14.05.1995

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
~~~~~~~
V2.19 (14.05.1995) :

 - sometimes produced garbage on uncompressed and CmpByteRun1 compressed
   files, which wasn't logically explainable (compiler-bug ?)
   Changed some parts for compressed files (works now), and bypassing
   this for uncompressed files by directly writing the single
   lines to disk
 - side-effect: uncompressed pictures now use much less memory than
   before when beeing written
 - again fixed some more things within the CAMG handling (should
   really accept "foreign" viewmodes now when reading)


V2.19 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed


V2.18 (25.03.1995) :

 - now fills in NativeDIPF of V2 GfxBuffers
 - BadCAMGs will no longer be rejected, if actually available
   (this happens, when 32 Bit ViewModes of GfxCards are set
    inside the CAMG chunk without any further flags...),
   also removed some 1.3-dependent validity checks, which might
   have caused trouble sometimes


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   : 2.15
 Release Date : 14.05.1995

 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
~~~~~~~
V2.15 (14.05.1995) :

- again fixed some more things within the CAMG handling (should
  really accept "foreign" viewmodes now when reading)

V2.14 (14.04.1995) :

- recompiled with SAS/C V6.55
- now saves A1 when opened/closed
- well, previous version was V2.13, not V2.18 (docs)

V2.13 (27.03.1995) :

- now fills in NativeDIPF of V2 GfxBuffers
- BadCAMGs will no longer be rejected, if actually available
  (this happens, when 32 Bit ViewModes of GfxCards are set
   inside the CAMG chunk without any further flags...),
  also removed some 1.3-dependent validity checks, which might
  have caused trouble sometimes

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

Version   : 2.18
Release Date : 29.04.1995

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 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
~~~~~~~
V2.18 (29.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed
 - (hopefully) fixed the last remaining problems with differing
   internal and pcx-defines "bytes per line" values.
   All PCX-brushes/PCC-files should now be read correctly.

V2.17 (11.03.1995) :

 - now rejects HAM

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   : 2.11
Release Date : 08.04.1995

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 (just like 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
~~~~~~~
V2.10 (08.04.1995) :

 - recompiled with SAS/C V6.55
 - added CPInfo-File
 - now saves A1 when opened/closed

V2.9 (05.03.1995) :

 - now fills in NativeDIPF of V2 GfxBuffers
 - also fixed initialization bug

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      : 2.15
 Release Date : 07.05.1995


 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


 There are also 68030(020) and 68040 versions of GIF.svobject, which
 can be found at "libs/680?0/svobjects".

 Bugs and other Problems
 ~~~~~~~~~~~~~~~~~~~~~~~~~
 Decoding is reentrant, encoding isn't:
 Please note, that the Encoding Routines of the Library are not (yet)
 fully re-entrant, so that only one Task may Encode a picture at a time.
 The Library itself manages it, that the other Task has to Delay() until
 the Encoding Routines are "free" again.
 This makes NOT REALLY A MATTER, because Encoding takes so much of the
 processor's time, that it is NOT EFFICIENT to ENCODE TWO OR MORE
 pictures AT THE SAME TIME ANYWAY !

```
History :
~~~~~~~~~
V2.15 (07.05.1995) :

 - fixed memory handling a little
 - saved some bytes of memory

V2.14 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed
 - added 68040 version

V2.13 (11.03.1995) :

 - now rejects HAM

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      : 2.14
Release Date : 14.04.1995

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
~~~~~~~
V2.14 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed

V2.13 (11.03.1995) :

 - now rejects HAM

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      : 2.2
Release Date : 14.04.1995

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
~~~~~~~
V2.2 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed

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      : 2.10
Release Date : 14.04.1995

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
~~~~~~~
V2.10 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed

V2.9 (11.03.1995) :

 - now rejects HAM

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   : 2.3
 Release Date : 14.04.1995

 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
~~~~~~~
V2.3 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed

V2.2 (11.03.1995) :

 - now rejects HAM

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   : 2.6
Release Date : 14.04.1995

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
~~~~~~~
V2.6 (14.04.1995) :

  - recompiled with SAS/C V6.55
  - now saves A1 when opened/closed

 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   : 2.8
 Release Date : 14.04.1995

 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
   ~~~~~~~
   V2.8 (14.04.1995) :

    - recompiled with SAS/C V6.55
    - now saves A1 when opened/closed

   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      : 2.11
   Release Date : 14.04.1995

   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    storage type

    None          1         1           (system)   1*(system)    BITPLANE
    None          1         8           1,3        8 / 24        ONEPLANE
    Mac PackBits  32773     1           (system)   1*(system)    BITPLANE
    Mac PackBits  32773     4           1          4             ONEPLANE
    Mac PackBits  32773     8           1          8             ONEPLANE
    LZW           5         8           1,3        8 / 24        ONEPLANE

    Not all of these modes may always work correctly. Please report any bugs.

   Writing :

    Depth              written as

    upto 256 Colors    Motorola (MM), None, 8, 1, ONEPLANE
    (24 Bit)           Motorola (MM), None, 8, 3, ONEPLANE
```

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

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

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.


History
~~~~~~~
V2.11 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed
 - added 68030 and 68040 versions

V2.10 (22.3.95) :

 - again, there was some debugging info
 - now rejects HAM
 - supports writing of 24 Bit data now

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

```
Version      : 2.2
Release Date : 14.04.1995

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 in in_ .
(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
~~~~~~~
V2.2 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed

V2.1 (30.12.1994) :

 - first version
```

1.29 Targa.svobject

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

Version      : 2.7
Release Date : 01.05.1995

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
  ~~~~~~~
  V2.7 (01.05.1995) :

   – recompiled with SAS/C V6.55
   – now saves A1 when opened/closed
   – when determing the best ViewMode, the OS would have returned
     a HAM6/8 ID for 24 Bit data. Now this is prevented the same
     way as inside the other SVObjects.

  V2.6 (11.03.1995) :

   – now rejects HAM

  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      : 2.5
Release Date : 14.04.1995

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
~~~~~~~
V2.5 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed

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   : 2.4
  Release Date : 14.04.1995

  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
  ~~~~~~~
  V2.4 (14.04.1995) :

```
- recompiled with SAS/C V6.55
- now saves A1 when opened/closed
```

```
V2.3 (11.03.1995) :
```

```
- now rejects HAM
```

```
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

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

```
Version      : 2.2
Release Date : 14.04.1995
```

```
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
~~~~~~~
V2.2 (14.04.1995) :
```

```
- recompiled with SAS/C V6.55
- now saves A1 when opened/closed
```

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

– first version

1.33 MAC.svobject

Version   : 2.2
Release Date : 14.04.1995

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
~~~~~~~
V2.2 (14.04.1995) :

 – recompiled with SAS/C V6.55
 – now saves A1 when opened/closed

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 FastILBM24.svobject

This one is a "bonus product", which has been written
by me and may only be installed, when needed.

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

                    Show Documentation

1.37 YUVN.svobject

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

 Version   : 2.4
 Release Date : 14.04.1995

 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
~~~~~~~
V2.4 (14.04.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed

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)

1.38 QRT.svobject

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

Version   : 2.1
Release Date : 14.04.1995

Description

```
~~~~~~~~~~~
QRT.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 QRT (POV RayTracer) Files.
In detail these are :

Reading :

 - QRT Dump 24 Bit

Writing :

 - QRT Dump 24 Bit

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

History
~~~~~~~
V2.1 (14.04.1995) :

 - first version with both, read- and write support
 - already compiled with SAS/C V6.55
 - already saves A1 when opened/closed
```

1.39   UtahRLE.svobject

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

Look out for an archive called "svoUtah22.LHA" or similar.
```

1.40   ECS.svdriver

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

Version      : 2.3
Release Date : 06.05.1995

Requirements
~~~~~~~~~~~~
- OS V2.04+ (V37)+ and its libraries
- superviewsupport.library V5+
- AMIGA with Enhanced Chip Set (ECS)
```

Description
~~~~~~~~~~~
ECS.svoperator is an external SV_GfxBuffer display module for
superview.library, which allows object-oriented displaying on
ECS displays.

This Driver supports the following :

Dimensions   Depth   Type

[ECS]   [ECS]   BITPLANE
[ECS]   8/(24)   ONEPLANE   (Chunky Pixel)

The 8-Bit mode will perhaps only work on ECS systems which have any
Graphic Card installed, which allows 256 or more colors in a way
of an Intuition emulation.
24 Bit Graphics will be displayed as "best guess of 256 colors"
if no SVOperator is specified.

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

ControlPad-Switches
~~~~~~~~~~~~~~~~~~~~
ControlPad-Name   : "ENV:superview-library/ECS.controlpad"
ControlPad-Commands : - 24BITOPERATOR=<OperatorName>
                        ; (case-sensitive, ".svoperator" may be added)
                        ; e.g. "24BITOPERATOR=24BitToHAM.svoperator"
                        ; or   "24BITOPERATOR=ExtractGrayScales"
                        ; specifies, which operation should be performed
                        ; on 24 Bit graphics _before_ displaying them
                        ; (if not specified displaying will fail)
                      - BITMAPCOPY=<DIRECT|RTG>
                        ; "BITMAPCOPY=RTG" prevents ECS.svdriver from
                        ; directly copying into Bitmaps, which will
                        ; result in a usage of more memory, but keeps it
                        ; working.
                        ; Default is "BITMAPCOPY=DIRECT".

Known Bugs
~~~~~~~~~~
ad 1)

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 ?!).

ad 2)

Do not try to display HAM6/8 images with older versions of the

```
   RetinaEmu-Software via ECS.svdriver.
   It'll sure crash.


   History
   ~~~~~~~
   V2.3 (6.5.1995) :

    - sometimes crashed on trying to open 24 Bit Screens (no longer tries)


   V2.2 (30.4.1995) :

    - recompiled with SAS/C V6.55
    - added CPInfo-File
    - now saves A1 when opened/closed
    - now supports a new switch (BITMAPCOPY=RTG), which enables it to work
      with more Workbench Emulations than before when displaying BITPLANE
      type GfxBuffers
    - fixed svsupport.library's version requirements (V1 did not suffice)


   V2.1 (25.3.1995) :

    - upgraded to version 2 specifications
    - removed some unused data from the handle structure
    - changed ViewMode validity check
    - added palette check for HAM6/EHB


****************************************************************************
* Revison history entries before V2.1 have been deleted for space reasons *
****************************************************************************
```

1.41 AGA.svdriver

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

   Version      : 2.3
   Release Date : 23.05.1995


   Requirements
   ~~~~~~~~~~~~
   - OS V3.00+ (V39)+ and its libraries (takes advantage of V40)
   - 68020+
   - superviewsupport.library V5+
   - 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, which allows object-oriented displaying on
   AGA displays.


   This Driver supports the following :
```

```
Dimensions          Depth          Type


[AGA]               [AGA]          BITPLANE
[AGA]               8/(24)         ONEPLANE    (Chunky Pixel)
```

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

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

ControlPad-Switches
~~~~~~~~~~~~~~~~~~~~
ControlPad-Name   : "ENV:superview-library/AGA.controlpad"
ControlPad-Commands : - 24BITOPERATOR=<OperatorName>
                        ; (case-sensitive, ".svoperator" may be added)
                        ; e.g. "24BITOPERATOR=24BitToHAM.svoperator"
                        ; or   "24BITOPERATOR=ExtractGrayScales"
                        ; specifies, which operation should be performed
                        ; on 24 Bit graphics _before_ displaying them
                        ; (if not specified, "best guess" colors will
                        ; be used, which is really slow)
                      - BITMAPCOPY=<DIRECT|RTG>
                        ; "BITMAPCOPY=RTG" prevents AGA.svdriver from
                        ; directly copying into Bitmaps, which will
                        ; result in a usage of more memory, but keeps it
                        ; working.
                        ; Default is "BITMAPCOPY=DIRECT".

Known Bugs
~~~~~~~~~~
ad 1)

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 ?!).

ad 2)

Do not try to display HAM6/8 images with older versions of the
RetinaEmu-Software via AGA.svdriver.
It'll sure crash.

History
~~~~~~~

V2.2 (23.5.1995) :

  - ViewPortExtra structure always had been free'd just right before
    the Screen has been closed. Possibly might have caused Gurus,
    when other active tasks did heavy memory allocation actions
    during the Screen was still open.
```

```
   V2.2 (30.4.1995) :

    - recompiled with SAS/C V6.55
    - added CPInfo-File
    - now saves A1 when opened/closed
    - now supports a new switch (BITMAPCOPY=RTG), which enables it to work
      with more Workbench Emulations than before when displaying BITPLANE
      type GfxBuffers
    - fixed svsupport.library's version requirements (V1 did not suffice)

   V2.1 (19.2.1995) :

    - upgraded to version 2 specifications
    - removed some unused data from the handle structure
    - changed ViewMode validity check
    - added palette check for HAM6/EHB


****************************************************************************
* Revison history entries before V2.1  have been deleted for space reasons *
****************************************************************************
```

1.42 EGS.svdriver

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

   Version     : 2.2
   Release Date : 08.04.1995


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

   The 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 ...

Programmers
~~~~~~~~~~~
Because of the nature of EGS no pointers to the real Screen or Window
(for IDCMP checking) are returned by the related library functions,
because EGS is not an Intuition compatible system.
To fix this a little bit, every time when displaying a graphic on the EGS-
Screen a Workbench Window (SystemRequester) is opened which allows usual
IDCMP accesses (and thus preserves your settings).
The actual EGS Window will lack any IDCMP-like flags.

History
~~~~~~~
V2.2 (8.4.1995) :

 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed

V2.1 (19.2.1995) :

 - upgraded to version 2 specifications
 - removed some unused data from the handle structure
 - made some improvements concerning speed (especially 24 Bit sources)
 - now opens SystemRequester instead of ugly window (will always fit ;-)
   Also updated this doc concerning the description of this behaviour.

****
* Revison history entries before V2.1 have been deleted for space reasons *

****

1.43 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

1.44 Retina.svdriver

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

Version   : 2.3
Release Date : 21.05.1995

Requirements
~~~~~~~~~~~~
- OS V2.04+ (V37)+ and its libraries
- superviewsupport.library V5+
- retina.library   V7+
- retinaemu.library   V1+
- AMIGA with Retina Z2/Z3 Graphics Card and Software

Description
~~~~~~~~~~~
Retina.svdriver is an external SV_GfxBuffer display module for
superview.library, which allows object-oriented displaying on
Amigas with the 16/24 Bit Retina Workbench Emulation System.

The corresponding Retina-libraries are needed.

This Driver supports the following :

Source   Depth   Special Modes   Dimensions   Dest. Depth

ONEPLANE 8/24   HAM6, HAM8   [Retina-Dependent] 16 / 24
BITPLANE 1..8   HAM6, HAM8   [Retina-Dependent] 16 / 24

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.
Also EHB graphics may not result in correct displays.

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

Credits

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

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


ControlPad-Switches
~~~~~~~~~~~~~~~~~~~
ControlPad-Name     : "ENV:superview-library/Retina.controlpad"
ControlPad-Commands : - EMUSCREENDEPTH=<16|24>
                        ; beginning depth for opening Retina Screens
                        ; Default is 24 Bit (if opening fails, it is
                        ; also tried to open a 16 Bit Screen, then)
                      - AUTOSCROLL=<ON|OFF>
                        ; Disables any Autoscroll features, which
                        ; results in STD_WIDTH for any Retina Screens
                        ; Default is ON.
                      - MAX_AUTOSCROLL_WIDTH=<Width>
                        ; If the picture is wider than this value,
                        ; we don't try to use autoscroll any longer.
                        ; You should set this value, if your
                        ; Retina has not much on-board RAM.
                        ; Default is: (no restrictions)
                      - MAX_AUTOSCROLL_HEIGHT=<Height>
                        ; If the picture is higher than this value,
                        ; we don't try to use autoscroll any longer.
                        ; You should set this value, if your
                        ; Retina has not much on-board RAM.
                        ; Default is: (no restrictions)
                      - PASSTHROUGH_MODES
                        ; If Retina should ever support own
                        ; ScreenMode definitions like "Retina:...",
                        ; you should enable this switch.
                        ; Default: (NOT SET !)

History
~~~~~~~
V2.3 (21.5.1995) :

 - does no longer install itself to the lists, when the retina*.libraries
   are missing and thus does not crash, if so.

V2.2 (18.5.1995) :

 - fixed small bug concerning oversized buffers (no more copying
   across window borders)

V2.1 (29.4.1995) :

 - first release (pre-released a V2.1ß as V2.0 on 29.04.1995)
```

1.45 CyberGraphics.svdriver

Version   : 2.2
Release Date : 23.05.1995

Requirements
~~~~~~~~~~~~
- OS V3.00+ (V39)+ and its libraries
- 68020 or greater processor

- superviewsupport.library V5+
- cybergraphics.library   V40+

- AMIGA with CyberGraphics System and appropriate Graphics Card

Description
~~~~~~~~~~~
CyberGraphics.svdriver is an external SV_GfxBuffer display module for
superview.library, which allows object-oriented displaying on
Amigas with the 16/24 Bit CyberGraphics Workbench Emulation System.

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

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.
Also EHB graphics may not result in correct displays.

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

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

Credits
~~~~~~~
Thanks to Ingenieurbüro Helrich, for supplying the PiccoloSD64 card.

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

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

```
History
~~~~~~~
```

```
V2.2 (23.5.1995) :
```

```
 - removed AGA WritePixelLine8() workaround with adjusted width
   and additional line buffer, also for WriteChunkyPixels() (V40)
 - under V40 we now do only a single call to WriteChunkyPixels(),
   not line-by-line (removed another AGA workaround)
 - removed any HAM6/8 related screenmode generation code, since
   these will be displayed in 16/24 bit
 - removed possible generation of AGA screenmodes, if no
   appropriate CyberGfx mode had been available (now will fail instead)
 - will no longer try to set palettes on opened 16/24 Bit Screens,
   when graphics is HAM6/8
 - does no longer request "interleaved BitMaps"
 - when searching for a screenmode, at least 256 colors are requested,
   since the BestModeID() variation of cybergraphics.library did
   not handle lower values very well sometimes.
   The Screen itself then will only opened with the desired number
   of colors.
 - now using original BestModeID() for 16/24 Bit depths, since it was
   possible to confuse the CG pendant to return an 8 Bit mode even
   if 16 Bit had been requested (did not return INVALID_MODE, but
   8 Bit mode instead). The OS function works fine instead.
 - removed attached ViewPortExtra, which might confuse CyberGraphics (?!)
 - added "SMALLSCREENS" keyword, which is absent by default, since
   CyberGraphics sometimes has problems with screens, which are
   _extremely_ smaller than 320x240. Since I do not know the exact
   values (may differ from GfxCard to GfxCard) I use the default
   minimum values of CyberGraphics Screenmodes instead.
   Set this keyword, if you are sure, that it will work, only.
 - should be quite faster now due to the changes
```

```
V2.1 (21.5.1995) :
```

```
 - first public release
```

1.46 PICASSO.svdriver

This one is a "third party product", which has been written
by Patrick Ohly.

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

Look out for an archive called "SVDPicasso.LHA" or similar.
On AmiNet:

    V1.0: SVDPicasso.LHA
    V2.0: SVDPicasso2_0.LHA

Any of the official SuperView Support BBSs should also supply a copy.

1.47 MERLIN.svdriver

This one is a "third party product", which has been written
by Thomas Eigentler.

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

Look out for an archive called "SVDMerlin1_1.LHA" or similar.
On AmiNet:

    V1.1: SVDMerlin1_1.LHA

Any of the official SuperView Support BBSs should also supply a copy.

1.48 XOR.svoperator

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

 Version   : 1.6
 Release Date : 30.04.1995

 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
~~~~~~~
V1.6 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - added CPInfo-File
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.5 (12.03.1995) :

 - was compiled for '020 previously. Fixed.

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

1.49   24BitToHAM.svoperator

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

Version   : 1.7
Release Date : 07.05.1995

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
~~~~~~~
V1.7 (07.05.1995) :

 - removed some garbage
 - revised memory handling


V1.6 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - added CPInfo-File
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (now returns version 2 GfxBuffer with NativeDIPF = DIPF_IS_HAM)


V1.5 (12.03.1995) :

 - was compiled for '020 previously. Fixed.


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
```

1.50 Crop.svoperator

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


Version      : 1.1
Release Date : 30.04.1995


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

Crop.svoperator may extract shapes of any size from any 2..256 Color
or 24 Bit graphics.
Values for "LeftEdge", "TopEdge", "Width" and "Height", describing
the crop box have to be supplied via ControlPad settings.

```
ControlPad-Switches
~~~~~~~~~~~~~~~~~~~
ControlPad-Name     : "ENV:superview-library/Crop.controlpad"
ControlPad-Commands : - CROP_LEFTEDGE=<Value>
                        ; crop from x position
                        ; (will be adjusted, if >= source width)
                      - CROP_TOPEDGE=<Value>
                        ; crop from y position
                        ; (will be adjusted, if >= source height)
                      - CROP_WIDTH=<Value>
                        ; crop how many x pixels from leftedge
                        ; (will be adjusted, if too large)
                      - CROP_HEIGHT=<Value>
                        ; crop how many y pixels from topedge
                        ; (will be adjusted, if too large)
```

```
History
~~~~~~~
V1.1 (30.04.1995) :

 - first version
```

1.51 Dither24Bit.svoperator

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

```
Version      : 1.4
Release Date : 07.05.1995
```

```
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 modified 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
~~~~~~~
V1.4 (7.5.1995) :

 - the color with the highest intensity will no longer be the first
   color, so that the first color will usually be black (if the
   graphics contained enough of it)
 - revised memory management

V1.3 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - added CPInfo-File
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.2 (12.03.1995) :

 - was compiled for '020 previously. Fixed.

V1.1 (26.12.1994) :

 - first version
```

1.52 HilbertDither256.svoperator

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

Version      : 1.5
Release Date : 30.04.1995

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
    ~~~~~~~
    V1.5 (30.04.1995) :

     - requests Dos, Intui, Gfx V37 and SVSupport V5 now
     - recompiled with SAS/C V6.55
     - added CPInfo-File
     - now saves A1 when opened/closed
     - fixed GfxBuffer version handling
       (sets correct version now; fixed NativeDIPF problems)

    V1.4 (12.03.1995) :

     - was compiled for '020 previously. Fixed.

    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
```

1.53   24BitToHAM.svoperator

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

 Version      : 1.6
 Release Date : 30.04.1995

 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
 ~~~~~~~
```

```
V1.6 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - added CPInfo-File
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.5 (12.03.1995) :

 - was compiled for '020 previously. Fixed.

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
```

1.54   AnyTo24Bit.svoperator

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

Version      : 1.5
Release Date : 14.05.1995

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
~~~~~~~
V1.5 (14.5.1995) :

 - it was still possible to get a 24 Bit graphics with a HAM ViewMode,
   which sometimes might have had unexpected side-effects
   (rejected by HAM-insufficient operators or dithered to 256 Colors
```

```
      but displayed in HAM mode, etc.)

 V1.4 (7.5.1995) :

  - revised and updated memory handling

 V1.3 (30.04.1995) :

  - requests Dos, Intui, Gfx V37 and SVSupport V5 now
  - recompiled with SAS/C V6.55
  - now saves A1 when opened/closed
  - did not recognize HAM, if ViewMode had been changed before
  - fixed GfxBuffer version handling
    (sets correct version now; fixed NativeDIPF problems)

 V1.2 (12.03.1995) :

  - was compiled for '020 previously. Fixed.

 V1.1 (09.02.1995) :

  - first version
```

1.55   ExtractGrayScales

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

 Version      : 1.5
 Release Date : 30.04.1995

 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
 ~~~~~~~
 V1.5 (30.04.1995) :
```

```
 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - added CPInfo-File
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.4 (11.03.1995) :

 - now rejects HAM
 - was compiled for '020 previously. Fixed.

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
```

1.56 ExtractRed

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

Version     : 1.4
Release Date : 30.04.1995

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
~~~~~~~
V1.4 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.3 (12.03.1995) :
```

```
 - now rejects HAM
 - was compiled for '020 previously. Fixed.

V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (29.09.1994) :

 - first version
```

1.57 ExtractGreen

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

Version      : 1.4
Release Date : 30.04.1994

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
~~~~~~~
V1.4 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.3 (11.03.1995) :

 - now rejects HAM
 - was compiled for '020 previously. Fixed.

V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (29.09.1994) :

 - first version
```

1.58   ExtractBlue

```
Version      : 1.4
Release Date : 30.04.1995
```

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
~~~~~~~
V1.4 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.3 (12.03.1995) :

 - now rejects HAM
 - was compiled for '020 previously. Fixed.

V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (29.09.1994) :

 - first version

1.59   TopToBottom

```
Version      : 1.4
Release Date : 30.04.1994
```

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
~~~~~~~
V1.4 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.3 (12.03.1995) :

 - was compiled for '020 previously. Fixed.

V1.2 (28.12.1994) :

 - fixed bug in (re-)initialization module

V1.1 (01.11.1994) :

 - first version
```

1.60 LeftToRight

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

Version      : 1.4
Release Date : 30.04.1995

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
~~~~~~~
V1.4 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.3 (12.03.1995) :
```

```
- now rejects HAM
- was compiled for '020 previously. Fixed.
```

```
V1.2 (28.12.1994) :
```

```
- fixed bug in (re-)initialization module
```

```
V1.1 (01.11.1994) :
```

```
- first version
```

1.61 Rotate

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

Version   : 1.3
Release Date : 30.04.1995

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
~~~~~~~
V1.3 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - added CPInfo-File
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.2 (12.03.1994) :

  - now rejects HAM
  - was compiled for '020 previously. Fixed.

V1.1 (02.01.1995) :

```
    - first version
```

1.62   Scale50

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

Version      : 1.4
Release Date : 30.04.1995

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
~~~~~~~
V1.4 (30.04.1995) :

 - requests Dos, Intui, Gfx V37 and SVSupport V5 now
 - recompiled with SAS/C V6.55
 - added CPInfo-File
 - now saves A1 when opened/closed
 - fixed GfxBuffer version handling
   (sets correct version now; fixed NativeDIPF problems)

V1.3 (12.03.1995) :

 - now rejects HAM
 - was compiled for '020 previously. Fixed.

V1.2 (25.02.1995) :

 - sometimes failed on downscaling odd-width/height graphics

V1.1 (02.01.1995) :

 - first version
```

1.63   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 !
  [9] ... as well as texts found on AmiNet, BBS or CD-ROM.
```

1.64 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).
```

1.65   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)
```