NotesAndHints

COLLABORATORS			
	TITLE : NotesAndHints		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		April 16, 2022	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

NotesAndHints

1.1 NotesAndHints

                    "Notes and Hints" FAQ - V11.11 (24.9..95)

            SuperView-Library sounds impressive, but...

            How much Memory does this program eat ?!

            What about Datatypes ?

            Problems with converting HAM Files

            Problems with displaying 24 Bit Files

            Problems with converting 24 Bit Files

            Programmers note for releasing SVObjects/Drivers/Operators

            One of the advertised features does not work...

            Philosophy... :-)

            Copyrights

1.2 usagenote

```
 SuperView-Library sounds good, but how shall I use it ?
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
   As with V11.8 superview.library will only be distributed together
   with programs, whose authors did get a license for usage and
   distribution. This may be Freeware, Shareware or commercial programs.

   For example, my program SuperView, which is SHAREWARE and also
   currently most appropriate Viewer-Program for the library, always
   comes together with the latest available Library-Version within
```

a subdirectory. You then just have to install to program itself
and also the library.

When using more than one program with superview.library-support,
take care only to install the most up-to-date versions and not
to overwrite them with older ones.

1.3 memprobs

How much Memory does this program eat ?!
~~~~~~
  Well, er, this is a good question, because there are no fixed
  limits ...
  superview.library and its attached SVObjects, SVDrivers and
  SVOperators request as much memory as they need - which directly
  depends on the size of the processed graphics - and there's no
  automatic virtual memory manager or such (I introduced some
  vmem-support it with V9.9, but upto now there aren't any modules
  making use of it).

  But actually nearly no CHIP memory is needed for most of the actions,
  so 1-2 MB CHIP mem should suffice for almost all actions
  (if you don't want to display graphics sized 2048x2048x8 with
   AGA - I mean, what do we talk about ?!).

  Most of the actions are performed in FAST Ram if possible, so you
  should have at least 2 MB Ram at all (1 CHIP, 1 FAST or 2 CHIP),
  better 4 MB (2 CHIP, 2 FAST) or more.

  The large memory usage results out of the buffering-technique, which
  is used in the superview-Library system.
  This technique increases memory usage by using large (full-sized)
  buffers, which often do exist twice e.g. for conversion from
  chunky-pixel to bitplane format or vice versa.
  But this increases SPEED !!
  There are perhaps no "real" multimedia-systems out there, which have
  less than 6 - 16 MB and why shouldn't we use this memory ?

  On a 68040/68060 system you should see superview.library flying ...

  OK, back to earth ! Here's what you can do to make more memory available
  to superview.library :

   - first of all, check if there are some background-programs running,
     which you do not really need (especially Commodities).
     Remove them !
   - call "avail flush" in the Shell or select "flushlibs" in the
     WB-debug menu (available when started with "LoadWB -debug").
     As with V4.2 SuperView also allows flushing Libraries via Menu calls.
   - decrease the "AddBuffers" values of drives, which you do not
     really need (e.g. DF2, DF3, PC0, ...)
   - last not least : close Workbench, stay in the Shell
   - start your favorite Viewer or Conversion program (e.g. SuperView)
   - again : call "avail flush"
     (this time also all unused SVObjects and SVDrivers will be flushed

out of the memory, so that we later only will have the needed ones
in Ram)

If you tried all of the above, and memory still does not suffice,
you should buy some more SIMMs at your local Computer Shop ...

1.4 datatypes

What about Datatypes ?
~~~~~~~~~~~~~~~~~~~~~~~~
Well, guess what: since superview.library deals with graphics files,
there might be no need to support other kinds of datatypes.
On the other hand, it would be nice, to have one viewer/player
for all the Datatypes and - maybe - for replacing the standard viewer.

The current implementation of Datatypes-Support inside superview-
library allows it to read and/or display ANY Picture-Datatype and to
link it - keeping all its functionality - into superview.library's control
 structures.

Putting the single keyword "ANYDATATYPES" into "ENV:SuperView-Library/
LIBRARY.controlpad" enables support for datatypes other than Graphics
(recognition of changes to this one does REQUIRE a re-initialization of
 the library, e.g. via reboot/reset or "avail flush" when not accessed).

superview.library (i.e. SuperView) will then be able to display ANIMs,
CDXLs, WAVs, and so on, if the according datatype is already in memory
(that's the most important part ;-)

This enables you to replace the standard DT-Viewer of the WB in
most of its facilities.

1.5 hamprobs

Problems with converting HAM Files
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 The Problem

  You have problems converting HAM6 or HAM8 files into GIF, PCX or
  such "alien" File Formats ?

 The Reason

  Well, this is because those FileFormats do not preserve any ScreenMode
  information, so that a HAM6 file would become just a plain 64 Color
  BitMap graphic and HAM8 turns into a 256 Color graphic.
  And this would look really UGLY without doing any further dithering,
  so the savers for non-viewmode-preserving fileformats just don't
  accept HAM data as input.

 The solution

```
   There are two possible ways :
   a) only convert to FileFormats, which preserve the ScreenMode information
      (currently there are only IFF-ILBM, IFF-ACBM and SVO).
   b) convert (and maybe dither) them before saving, e.g. do a 24 Bit
      conversion via "AnyTo24Bit" and then - maybe - dither them down
      to 2..256 colors via "Dither24Bit"
```

1.6   24bitprobs2

```
Problems with displaying 24 Bit Files :
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 The Problem
 -----------
  "All of my attempts in displaying my really nice 24 Bit graphics with
   AGA.svobject resulted in really ugly colors. What shall I do ?"

 The Reason
 ----------
  You did not set an available or valid 24 Bit SVOperator, which
  AGA.svdriver - or as well ECS.svdriver - could use for adjusting
  the numer of colors to less than 256.

 The solution
 ------------
  Within AGA.controlpad (respectively ECS.controlpad) you may set
  "24BITOPERATOR=<operatorname>" and specify a specific operator
  for e.g. dithering 24 Bit graphics to 256 colors, converting
  them to grayscales or HAM6/8 etc.

  The same trick may take place for ECS.svdriver on systems, which
  can't display more than 16/32(64EHB) colors, so that specifying
  the (optional) "8BITOPERATOR=<operatorname>" would allow automatic
  conversion to less colors.


  For AGA-only systems it might also be useful to adjust the output
  ot SVObject to AGA needs wherever possible.
  For example the JPEG.svobject does allow on-the-fly dithering
  which may be activated via the switch "COLORDEPTH=8" within
  JPEG.controlpad. Output then will always be done in 256 Colors
  (or grayscales) only.


  The alternative to always do the dithering by hand via
  calling SVOperators from SuperView's GUI also is another
  possible way.
```

1.7   24bitprobs

```
Problems with converting 24 Bit Files :
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
 The Problem
```

```
        -----------
  You have problems converting from one 24 Bit file format into
  other 24 Bit file formats ?

 The Reason
 ----------
  Not necessarily all SVObjects, which support reading of 24 Bit Files do
  also support writing of 24 Bit Files.
  SVObjects, which support writing 24 Bit data yet are:

   ILBM         JPEG          Targa
   BMP          SunRaster     YUVN
   PCX          FBM           PNM
   SVG          SGI           DEEP
   TIFF         QRT

 The solution
 ------------
  A temporary solution might be to use other 24 Bit programs for
  conversion, if you don't want to use one of the file formats, which
  are already supported.
```

1.8   prgnote1

```
 Programmers note for releasing own SVObjects/Drivers/Operators:
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
  The Idea
  --------
   When writing your own SVObjects/Drivers or Operators, and distributing
   them via Nets or MailBox/BBS, please name your archives in the follwing
   way :

           "svo*..*xy.EXT"   for SVObjects
    or     "svd*..*xy.EXT"   for SVDrivers
    or     "svp*..*xy.EXT"   for SVOperators

   where - "*..*" stands for the library's name          (e.g. "JPEG")
         - xy stands for the version and revision number (e.g. "22")
         - EXT stands for the archiver's extension        (e.g. "LHA")

   The Reason
   ----------
   The space for describing files on BBS is not very large (let's say
   about 20 characters) and most people do not really read the attached
   mails or readme-files usually.
   So it might be useful to see right from the FileName :
   "Yes, it's a SVObject" (svo#?) or "Yes, it's a SVDriver" (svd#?)
   and so on.

   An Example
   ----------
   For example, version 2.2 of my JPEG.svobject had been distributed as :

        "svoJPEG22.LHA"
```

        (now it is integral part of superview.library's "Bonus" directory,
         but there may still occur separate archives containing single modules
         from time to time)

1.9 bugnote

One of the advertised features does not work ...
~~~~~~
Well, sometimes it happens, that the elimination of one minor bug
causes two major bugs, so that a version of a SVObject or SVDriver
is actually much less useful, than previous versions, although a bug
has been fixed.

If you sometimes face such a problem, just install the old version of
this specific SVObject or SVDriver and keep the rest of the new
installation intact. SVDrivers and SVObjects are fully compatible
from the first version upto the current version.
Of course there might be exceptions, but those would be again Bugs ...

If you do not have any older versions to replace an existing SVObject
or SVDriver, you have to decide, whether you'll remove it completely
and just wait, until the next version comes out, or if you want to
use the buggy one nevertheless.

BTW, if you'll decide to report the bugs to me, they'll
perhaps be eliminated sooner ...

Example: (1) In V9.1-9.4 of superview.library there was a slightly buggy
              AGA.svdriver (autoscroll deactivated), which resulted out of
              an other bug-fix.
              Three possible solutions: a) using ECS.svdriver
                                        b) using old AGA.svdriver
                                           (with old bug)
                                        c) waiting for V9.5 ...

          (2) In V11.8, JPEG.svobject would have crashed the whole
              system at library startup-time when beeing installed
              (<= 68030 systems only).
              Solution: Since it still was located within the bonus dir,
                        it would have sufficed not to install it and
                        just continue using the old version (bug has been
                        fixed as with V11.9).

1.10 philosphy

   "Bei Risiken und Nebenwirkungen essen Sie die Packungsbeilage
    und @$_!@*# Sie Ihren Arzt oder Apotheker"

   (I do not feel able to translate this ^ into any other languages ...
    It's just a parody on a slogan, which has to be placed at the end
    of german commercials, if they offer pharmaceutical products, which
    might cause physical problems for the consumer ...)

```
Murphy's laws of working with BitMapped Graphics:
(Andy's release ;-)

 (excerpt)

 No. 23 : When you've integrated the fastest, smartest C routines
          for Huffmann-decoding, which you were able to find in the
          whole public domain, you'll see, that there's someone out
          there, which wrote a 20 times faster version in 68030 assembler.

 No. 34 : If you try to send files via UUCP, they'll be bigger than
          20 K with a likelihood greater than 99.999%.
          But only, if you don't see an other way to send them.

 No. 41 : If you make backups you'll overwrite them with the next
          buggy version, nevertheless.
          Guess ?

 No. 55 : After 20 nights and days of un-interrupted programming
          it'll happen, that in the 21th night a mysterios bug
          appears, which will make all of your previous work obsolete.

 No. 58 : Never believe, that the mysterious behaviour of your program
          might not necessarily be a "real bug".
          It's ALWAYS a "really big bug".
          Even bigger than you ever thought one single bug might be ...

 No. 64 : If you're making great usage of the OS and you've integrated
          all features, which this version offers to the programmers,
          it will take about half an hour and all BBS will post the
          message, that the new OS 999.1 is out and that it contains
          support for almost any BitPlane graphic formats in the
          whole wide world.

 No. 77 : If you've converted all your pictures to an other FileFormat
          - of course without making copies, 'cause you'd like to save
          a lot of diskspace - you'll detect that the Writer program of
          the FileFormat has been slightly defective, so that the
          supplied Reader program is the only one, which will read your
          pictures then, any longer.

 No. 87 : Ever tried to port some code from XX-DOS ?
          Still working on it ?
          Never mind, it's the best way to waste your time - really !

 No. 99 : If you've written a program, that writes BitMaps, you will
          perhaps have forgotten that sometimes the data has to
          be padded to giga-word boundaries, to make reading of
          the data faster on the new H.A.L. 5000.
```

1.11 copyrights

Copyrights

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