

**EGSAPIus**

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

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

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

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>EGSAPlus</b>	<b>1</b>
1.1	EGSPlus . . . . .	1
1.2	copyright . . . . .	1
1.3	requirements . . . . .	2
1.4	einführung . . . . .	2
1.5	install . . . . .	3
1.6	egs . . . . .	3
1.7	thanx . . . . .	3
1.8	zukunft . . . . .	4
1.9	probleme . . . . .	5
1.10	autor . . . . .	5

## Chapter 1

# EGSAPlus

### 1.1 EGSPlus

```

                                EGSPlus
Workbench Enhancement for the EGS Emulation

~ Copyright      Copyright
~ Requirements   What's needed
~ Introduction   What is EGSPlus
~ Installation   Installation
~ About EGS      What is EGS

~ Thanks
~ Future         Is there a Future ?
~ Problems       Bugs in EGSPlsu

~ Author         Who is responsible for this.
```

### 1.2 copyright

COPYRIGHT

The files in this distribution are Copyright ©1996 Jürgen Schober.

This Software is Freeware. It may be copied and spread wherever you want.

EGSPlus and the egspplus.library (c)Copyright 1996 Jürgen Schober

EGS is (c)Copyright 1990-94 Viona Development !

Installer and Installer project icon  
(c) Copyright 1991-93 Commodore-Amiga, Inc. All Rights Reserved.  
Reproduced and distributed under license from Commodore.

Installer 43.3 and Installer project icon  
(c) Copyright 1995-96 Escom AG. All Rights Reserved

---

## Disclaimer

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDER AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## 1.3 requirements

### System Requirements

- Amiga with OS 3.x (graphics.library V40)
- ~CPU 68020+
- ~EGS System 7
- ~EGSWB Version 7.039

### Recommended:

- ~WBStartUp+ to boot up EGSPlus asynchronously.
- NewIcons to see my icons =:->

## 1.4 einführung

### Introduction

Recently I found an interesting statement in an aminet program, it was the ODSys I think, which goes like this: '...the performance of the display is dependend on the implementation of the WritePixelFormat8() routine of your Intuition emulation...'. I thought, well, could not be a problem for my emulation since my gfx board is also organized in chunky mode, so transferring chunky data to the board should not be a problem... But when I tried it, it wasn't really useable. Even the 060 couldn't help here! I thought there must be a problem in the implementaion of my emulated routine. After some further investigation, the result is EGSPlus. EGSPlus consists of some new functions that replace the old Amiga OS 3.x functions that can deal with chunky data. The new functions are 100% EGS style guide compatible and will also work on my A2410 board

---

(this board isn't even mapped into the Amiga Memory).  
Now, xanim get 30-60fps in an WB window on playback speed (\*.mov,\*.mpg).  
...well with an 060, but I had 15-30 fps on an EGS screen with an 030  
before. This speed should also be reached under Intuition now.

Some technical:

The functions I have replaced are:

- ~WritePixelArray8()
- ReadPixelArray8()
- WritePixelLine8()
- ~ReadPixelLine8()
- WriteChunkyPixels()

The original code isn't called any more if the rastport lays on the EGS Screen. The patch contains two parts, the "EGSPlus", the loader program, and the "egsplus.library", the library that replaces the graphics.library calls. EGSPlus should be started in the WBStartup(+) drawer but can also be executed on runtime (gives a nice effect if you are running an animation while executing EGSPlus :-)  
All parts are written in 100% pure C code, no assembler has been used at all (to you assembler freaks out there :-). So this code (and future versions) are portable.

## 1.5 install

Installation

There is not much to say about the installation.  
Copy the egsplus.library into the libs: directory and the EGSPlus to a location of your choice (recommended is the WBStartUp(+) drawer) where it could be executed.

If you do not want to do this by your own, use the included installer script.

## 1.6 egs

The Enhanced Graphics System (EGS)

To use EGSPlus an installed EGS system is required. I won't talk much about EGS because if you use EGSPlus, you have to have the EGS anyway and I think you know what EGS is. Just that EGS was one of the first intuition emulators available. It does have a great concept, but isn't really supported any more. EGS is IMO very overloaded which makes it really hard to control by the Joe Average User.

## 1.7 thanx

---

And thanx goes to...

Special Thanks to Uli Sigmund, from whom I got some sources which made this work possible.

Thank you Brian King for your help in the english documentation.

Also I want to thank all the Amiga Shareware programmer out there that keep the Amiga alive and from whom it is possible for me to learn a lot about Amiga programing.

## 1.8 zukunft

Is there a Future ?

I have to say that in the end EGSPPlus was a garbage product. I was working on my own Intuition Emulation for a while when I came to the point to write the emulation code itself. The project I am working on is my eXtended Graphics(tm, short x-graphics) project, which should be a totally new emulation conception (the name might change though).

Well, the x-graphics is a real huge project. I just wrote the aos-emu API and didn't do much emulation code itself. Most time was used for the low level class handler routines so far. When I wrote the EGSPPlus stuff I found that my timetable to show some results by end of 96 will not hold. On the other hand from the programmers point of view EGSPPlus is a great success that shows that I am on the right way. I have decided to make my x-graphics modular. This means after EGSPPlus there will be a cybergraphics.library that will run under EGS (such a library was part of the x-graphics). On the first release this library will only support 8 bit screens, since the current EGS emulation does not support Intuition Screens with depth greater than 8 bits (256 colors). This leads to the next step which will be a totally new emulation for EGS which should also allow 15/16/24 bit screens under EGS/Intuition (maybe virtual 24 bit would also be possible). After all and because I cannot change and increase the power of EGS the whole Emulation code will be ported to my x-graphics low level engine. This is also neccessary, because we have to go to 3D graphics in the 90s. Intuition Level 3D graphics is part of the x-graphics conception.

Also, it might happen that I release new version of EGSPPlus since there are some other things that could be done for better intuition emulation. To rewrite the AllocBitMap() function would be one of this things, also the Cricle routine was a point to look into (e.g. the EGSTurbo fixes this...but is not fully EGS compatible), and at least the look into the layers.library would be worth a penny. It looks to me that regions are not always friends to the RastPort->BitMap which generates more overhead. EGS could do a good deal here with its superior memory managment, though to rewrite the layers will be a bigger part...(but it is neccessary anyway).

So, well, you see I have a lot to do. So I should stop talking now.

---

## 1.9 probleme

Problems

That's a good question. I haven't tested EGSPlus for a long...since it works so fine :-) But if you have any bug reports, just send me a note.

But... EGSPlus uses EGS calls at the lower level. EGS is a stack based system and because of that it really needs a lot of them - usually about 30000 bytes. It could happen, that Intuition programs get trouble with the stack usage now. I can't say this for sure, since I run the Workbench on my 2410 board which has troubles itself. Also I have a lot of patches running and finally I have a 060 in my machine...so...hm  
To increase the stack frame temporary while the function is executed would be a solution for that in the future.

For now I can only recommend to set the stack frame for programs that use these function to a higher value (e.g. 30KB) - but you have to check this out and determine if it is absolutely necessary.

Programs that use these functions include:

- ~xanim        now rocks on my SD64 /w 060 =:->
- xv2.18       now refreshes window redraws in realtime even on a 2410
- ~MultiView (the akJFIF.datatype)

## 1.10 autor

Who did this stuff

Standard Mail:

Jürgen Schober  
Muchargasse 35/1/4  
A-8010 Graz  
Austria/Europe

Internet:

e-mail : point@sbox.tu-graz.ac.at (private)  
         jschober@campusart.com (business)  
http:    <http://www.sbox.tu-graz.ac.at/home/point/Welcome.html>  
         <http://www.pointdesign.campusart.com/>

Machine:

A3000/060/2+12+16MB/1.2GB, EGS on a2410 & SD64

---