AppCon

COLLABORATORS			
	TITLE : AppCon		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		July 20, 2024	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

AppCon

1.1   AppCon Documentation - Contents

```
**********************************************************************
**                                                                  **
**                        AppCon 39.169                             **
**                                                                  **
**                      F R E E W A R E !                           **
**                                                                  **
**            An AppWindow handler for the shell                    **
**                                                                  **
**            Copyright © 1996 Stephan Fuhrmann                     **
**                                                                  **
**********************************************************************

   Introduction                        Why AppCon?
   Copyright                           Most important information.
   System Requirements                 Read this before usage.
   Usage                               How to use it.
   History                             What happened to this program.
   Author                              How to contact me.

   Bugs                                How to find and report bugs.


  Jetzt aber Shell!!
  Shell (tm)
```

1.2   AppCon Documentation - History

```
v37.101: first release
v37.105: improvement - when pathnames contain spaces
                       they are enclosed with quotation marks
v37.129: bug fix - forgot to free some memory
         improvement - using exec memory functions
         improvement - now waiting until the window is really activated
         improvement - using input.device rather than console.device to send
                       events following the recommendation of CBM
```

```
v37.160: improvement - AppCon detects now automatically the closing of
                        the shell and exits quietly - AppCon now is 100%
                        user friendly!
                        bug fix - a problem caused memory loss, fixed
v37.177: improvement - cleaned & optimized up the code
v38.16:  found a dangerous bug in OS 3.0, included a nice bypass
v38.18:  names with a trailing '+' are enclosed with quotation marks, too
v38.19:  recompiled with SAS/C 6.51
v38.46:  replaced StdIO with DosIO (smaller code), autoremove existing AppCon
v38.47:  some code optimizations (thx Teijo!)
v38.71:  removed C-startupcode and replaced by my own, shorter code
v38.73:  inverted the quotation mark feature by fault - fixed
v39.168: improvement - AppCon should be residentable (pure) now
         bug fix     - AppCon is now allocating it's own stack
                        this caused problems when 'run' started
                        AppCon with a stack of 3200 bytes
         improvement - AppCon uses commodity library instead of going directly
                        over the input.device - faster speed and higher reliability
         improvement - Cleaned up the code
         improvement - AppCon doesn't need to be RUNned anymore - it detaches  ↩
            itself
                        from the Shell if needed with the proper system routines
         distribution- removed the german documentation and changed
                        everything to AmigaGuide ®
         distribution- AppCon is FREEWARE now
         distribution- the sourcecode is not included anymore
                        recompiled with SAS/C 6.56
v39.169: bug fix     - AppCon now uses the proper way to find out whether
                        it shall detach itself from the Shell or not
```

1.3 AppCon Documentation - How to report a bug

Best is to read this section before discovering a bug.

Don't panic. Many bugs are harmless, most are never discovered. Be
sure to save all projects in other applications before running software
that is suspected to be buggy.

Is it really a bug?

This is a serious question. In some circumstances, other software crashes
and it appears that this application has a bug. Try to reconstruct the
same situation where the error occured again without any 3rd party
software running.

Out of that, check out if you're really right with your expectations. If
the software crashes, it's propably a bug. But if the software does
something that you didn't expect, it could be that you're not using the
software in the means the author has specified. Carefully reread the
sections covering your problem in the user manual to verify your actions.

Gosh - it is a bug!

Now that you have found out that you have discovered a real bug,
try to find out in which situations it occurs. Does it depend on other

software running simultaneously to the application? Does it appear in
certain situations? Does it depend on your system hardware? Find this out
by running this software on other machines than yours or by removing
hardware components (only do this if you know what you're doing).

How to report a bug

First, you should report the versions of all software you're running when
getting the bug. Also, your complete hardware specifications are helpful.

Describe exactly what you have done to come to the situation where the bug
occurs. Preferences or tooltype settings are important, too. Try to do
step-by-step instructions how to get to the bug. The author must be
able to reconstruct the bug on his machine. What happens when the
bug occurs? Can you come up with a guru/software error number?

If you've been using software to find the bug, send it included with
the bug report.

If the bug only occurs in 'extreme' situations, try to find the point
where the software doesn't work correctly anymore.

Other material like hex-dumps, screen dumps etc. may be interesting, too.

If you have finished your bug report, finally send it to the
author of the software. While electronic mail is faster,
snail mail on paper has a higher 'consistence' and can be read while
the computer of the author is crashing. Have this in mind when choosing
between email and snail mail.

Finally, always include your physical and electronical mail adresses, even
if you send snail mail.

1.4   AppCon Documentation - Requirements

Hardware
Amiga

Software
OS 2.0 or higher

exec.library
dos.library
commodities.library
intuition.library
workbench.library
keymap.library
utility.library

1.5   AppCon Documentation - Copyright

AppCon is a program that should improve the look & feel of the
Amiga and it's Shell.

AppCon may be included in Fred Fish's AmigaLib, on the AmiNet
and on Meeting Pearls.

1.6   AppCon Documentation - Introduction

The Amiga has got two user interfaces: the Workbench and the Shell.   The
advanced user has got to switch between these user interfaces very
often because each one has got its advantages and disadvantages.

When I switched from Workbench to Shell once again I had the idea that my
work could be much simpler when I could drag icons into the Shell-window and
have their names displayed in the command line. This is exactly what
AppCon does.

It declares the actual Shell-window as an AppWindow and lets you drop your
icons in this window. Then the name and path of the icon(s) are inserted
into the current command line exactly as if you typed them with your
keyboard, but slightly faster!

Created using SAS/C 6.56.

1.7   The AppWindow

An AppWindow is a facility provided by the Workbench. AppWindows let the user
drop Workbench icons into themselves and perform certain actions on each icon
and/or associated files.

1.8   AppCon Documentation - Usage

AppCon must only be used from the Shell.

AppCon can be RUNned from the Shell, but it automatically detaches
itself from the Shell if you don't explicitly RUN it. The best place to
call AppCon is your shell-startup. You've got to run

AppCon for *EVERY* Shell-Window seperately. Since AppCon is
residentable, this is not a big problem.

If you drop multiple icons at once into your Shell-window the names will be
seperated with spaces, names containing spaces are enclosed with quotation
marks.

When the shell exits and AppCon is the only program that's using the
Shell-window, AppCon will detect this and exit nicely, too.

AppCon periodically checks whether its Shell window is still being used.
You can modify the time between two checks by setting the TIMEOUT-option in
the command line of AppCon. When using AppCon, you'll see that with a
higher TIMEOUT the Shell window will react slower on closing commands, and
with a lower TIMEOUT AppCon will consume more CPU ticks. The TIMEOUT is in
tenths of a second, this means that a timeout of 10 (default) will force
AppCon to check every second whether the shell is still in use.

Simply start AppCon again to remove it from the current shell.

Example Use:

1> AppCon TIMEOUT=12

1.9   AppCon Documentation - Author

My address is:

Stephan Fuhrmann
Ostmarkstraße 19
76227 Karlsruhe
GERMANY

email:

Stephan.Fuhrmann@stud.uni-karlsruhe.de

email is usually being answered faster than snail mail.

Bank account

Commerzbank Schleswig
BLZ 210 400 10
Kto-Nummer: 8226516