

AView

Guido Mersmann

COLLABORATORS

	<i>TITLE :</i> AView		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Guido Mersmann	January 2, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	AView	1
1.1	AView	1
1.2	Introduction	2
1.3	System requirements	2
1.4	Feature List	2
1.5	Distribution	3
1.6	Restrictions of this unregistered version	3
1.7	How to register ?	4
1.8	Quick start	4
1.9	How to Install AView	5
1.10	AView	5
1.11	Future	6
1.12	History	6
1.13	Thanx	8
1.14	Buglist	8
1.15	The Author	9
1.16	Other programs !	9
1.17	Technical information about AView	10
1.18	The File Identification Commands	10
1.19	Command: MATCH	11
1.20	Command: FIND	12
1.21	Command: NAME	13
1.22	Command: FILESIZE	14
1.23	Command ASCII	14
1.24	Command: AND	15
1.25	Command: OR	16
1.26	Command: NOT	17
1.27	COMMAND: ()	18
1.28	Pattern: Name/M	19
1.29	Pattern: All/S	19

1.30	Pattern: ARG=To=From/K	19
1.31	Pattern: SetComment=SC/S	20
1.32	Pattern: ShowDetect=SD/S	20
1.33	Pattern: Debug/S	21
1.34	The AView preferences	21
1.35	Gadget: Sort by	21
1.36	Gadget: Name	22
1.37	Gadget: Select	22
1.38	Gadget: Delete	22
1.39	Gadget: All	23
1.40	Gadget: New	23
1.41	Gadget: Save	23
1.42	Gadget: Use	23
1.43	Gadget: Abort	23
1.44	The Edit Window	23
1.45	Gadget: Name	24
1.46	Gadget: Class	24
1.47	Gadget: Sub Class	25
1.48	link	26
1.49	Gadget: Stack	26
1.50	Gadget: Command	26
1.51	Gadget: Current Type	27
1.52	Gadget: Current Directory	27
1.53	Gadget: WB Output File	27
1.54	Gadget: CLI Output File	28
1.55	Gadget: Identifier	28
1.56	Gadget: Default %a	28
1.57	The File Type Selection	29
1.58	The Settings of AViewPrefs	29
1.59	Gadget: Expert Mode	29
1.60	Gadget: Save	30
1.61	Gadget: Use	30
1.62	Gadget: Abort	30
1.63	The menus of AViewPrefs	30
1.64	Menu: Open	31
1.65	Menu: SaveAs	31
1.66	Menu: Quit	31
1.67	Menu: Reset to Default	31
1.68	Menu: Last Saved	31

1.69 Menu: Restore	32
1.70 Menu: AView Prefs	32
1.71 Frequently ask Questions	32
1.72 AView and AViewPrefs aren't running	32
1.73 The online help isn't working! Why?	33
1.74 The online help is english only.	33
1.75 AViewPrefs sometimes is very slow.	33

Chapter 1

AView

1.1 AView

AView V1.35

(C) Guido Mersmann in 1996-1997

AVIEW IS A SHAREWARE PRODUCT

READ THE DISTRIBUTION SECTION FOR INFORMATION ON DISTRIBUTION

Contents

~~~~~Introduction~~~~~

~~~~~System~Requirements~~~~~

~~~~~Feature~List~~~~~

~~~~~Distribution~~~~~

How To Register

~~~~~How~To~Install~~~~~

AView

Quick start

AView

AView\_Prefs

AView command list

~~~~~FAQS~~~~~

~~~~~History~~~~~

```
~~~~~Future~~~~~
~~~~~Bug~List~~~~~
~~~~~Thanx~~~~~
~~~~~The~Author~~~~~
      Bug Reports,Incitements...

      My other amiga projects !
```

## 1.2 Introduction

Introduction  
-----

AView is an multiview enhancement. You ask why you need AView? Every file needs a datatype. Many datatypes aren't available or not very effective.

You want to view a JPG picture and your preferred viewer? No problem! AView allows to specify a viewer for every file format.

AView detects your file and runs the specified program or tool.

## 1.3 System requirements

System requirements  
-----

AView (should work) works on all Amigas with OS 2.0 or higher.

Some functions are not available under 2.0, because they presume OS3.x+ (MemoryPools,...)

From OS2.1 on AView is able to work multilingual.

If you want to run programs by using the WB-Mode you need the WBStart.library V2.x. This library is under development of Stefan Becker and is available on Aminet.

That`s all!

## 1.4 Feature List

Feature List  
-----

\textdegree{} Local support

---

\textdegree{} Kickstart 2.04 or higher is required.

\textdegree{} Programmed in Commodore Styleguide

\textdegree{} All adjustments are possible per Gadtools-GUI

\textdegree{} Different versions for different Kickstart versions.

\textdegree{} 100% Assembler

\textdegree{} Online help

\textdegree{} pre defined file types for an easy configuration

\textdegree{} allows to specify own file types.

## 1.5 Distribution

Distribution

-----

This program should be published on every public medium, as long as all files in the archive are unchanged. If this medium is a CD I expect a free copy. (AminetCDs excluded)

AView is Shareware. The unregistered version of AView isn't able to use more than 10 file types.

I expect from everybody, who uses AView regularly, 10 DM or the same amount in another currency. I think this does not hurt anybody, and I will have more fun to integrate new features or write a new game.

Updates may be requested from Aminet and my local support box "The kiteman" 2:2449/246

I am not liable for injures or data loss caused by AView. The use of AView is on your own responsibility !!!

## 1.6 Restrictions of this unregistered version

Restrictions of this unregistered version

-----

The unregistered version of AView is not able to process more than 10 file types.

This should be enough for your test procedure.

The full version allows to configurate unlimited file types.



## 1.7 How to register ?

How to register ?

-----

Send

me  
following things:

\textdegree{} Shareware fee about 10DM  
\textdegree{} one floppy disk  
\textdegree{} Your personal data (for the keyfile).  
\textdegree{} An envelope with stamps for about 2DM.

And you'll get the latest AView version including a keyfile.

People, registered before keyfiles were introduced, need not send money. (-8

You needn't send me a disk and an envelope, if you send  
me

20 DM or  
transfer it to my account. I think this is quite fair. I want to  
program and I don't want to buy envelopes, stamps and disks. And from  
20 DM won't be a lot left for me.

I don't make any exceptions concerning sending back disks. If you don't  
send an envelope or a disk, you are registered, but you don't have  
anything of this. you can come personally and get your disk or you must  
pay more.

If you'd like to get your key file via internet or fido, please send  
the money and your PGP key. Via Fido net I only send key files within  
Germany (Crash).

IMPORTANT:

I need your full address with christian and surname, your street and  
your living place and very important the name of the program you want  
to register.

## 1.8 Quick start

Quick start

-----

After installing AView you should run AViewPrefs. Some file types are  
installed by default and are ready to be modified.

If you want to include new types, then use the  
select  
gadget. A new

requester  
 opens and allows to choose one of the pre defined file  
 types.

After selecting one listview in the  
 main  
 window you are able to open  
 the

Edit  
 window by using the "Return"-Key. It is also possible to  
 choose via double clicking on one entry.

The  
 Edit  
 window allows to specify the command. This command will be  
 started any time AView recognizes the specified file type. Use  
 %f  
 to  
 define the position of the file name. AView replaces this pattern  
 automatically during operation.

Repeat this actions until all file types are defined.

Note: The  
 unregistered  
 version is not able to handle more than 10 file  
 types.

## 1.9 How to Install AView

How to Install AView  
 -----

To install AView please use the original Commodore Installer. It  
 creates a directory, in which AView is installed.

If you do not like AView, you can delete the whole drawer to destroy  
 AView completely.

## 1.10 AView

-----  
 AView

This is the main program and needed for daily usage. You are able to  
 run AView via CLI, Workbench or AppIcon, by using Toolmanager or  
 something like that.

Run via CLI  
 -----

This is the aview template:

```
"AView
    Name/M
    ,
    ARG=To=From/K
    ,
    All/S
    ,
    SC=SetComment/S
    ,
    SD=ShowDetect/S
    ,
    Debug/S
"
```

Run via Workbench

-----

Under normal conditions it is not useful, because AView isn't able to get file names. AView opens a ASL-file requester.

Toolmanager and ..

-----

By using tools like Toolmanager or something like that you are able to put AView on Docks, AppIcons, Menus and many more. AView will view all files without any problems.

But it is required to define your preferred file types . This procedure is required only once.

## 1.11 Future

Future

-----

\textdegree{} Icon Replacement. AView will replace icons by a specified default.

\textdegree{} I don't know. (-8 Your wishes are welcome.

## 1.12 History

History

-----

This text is telling the story of AView like a diary.

---

- 0.95B First working beta version.
  - 0.96B Small changes.  
Manual updated and AView adjusted to allow the new features of the preference window
  - 0.97B Small CurrentDir bug fixed. This bug causes crashes when using "mfs-handler" and "AmiCDFS". It seems that SystemTagList is doing some strange things by using the "NP\_Current" tag that causes such problems. "df0:" it self didn't make any problems. AViewPrefs is sorting all entries automatically.
  - 0.98B Small Match bug fixed.  
Debug mode implemented.  
Small prefs changes.
  - 0.99B Some internal adjustments.
  - 1.00 Small bugs fixed.  
  
    ShowDetect  
        feature implemented. (requested by Andreas Boerner)  
Empty files caused problems. (reported by Andreas Boerner)  
France catalogs included. (Thanks to Jerome Chesnot)  
Enlarged string gadgets. (requested by Andreas Boerner)
  - 1.01  
    NOT  
        Command implemented. (requested by Andreas Boerner)
  - 1.02 Small sourcecode optimizations.
  - 1.03 "1" and "0" are now valid commands, so it is possible to use AView to solve mathematical problems. I needed this "feature" to check the internal commands without using test files for recognition.
  - 1.04 Some Manual adjustments.
  - 1.05 It is possible to define Lha if DMS is already defined.  
(reported by Andreas Boerner)
  - 1.06 New Command  
    FILESIZE  
        . It allows to detect a file by its size.
  - 1.07 BUGFIX: No longer endless loops. Thrust me! 5 hours of intensive debugging. It wasn't possible to debug this direct, because the bug wasn't reproducible. I had to debug a running task.  
(reported by Andreas Boerner, Silke Bormann and me)
  - 1.08 Kickstart-Types implemented and  
    ShowDetect  
        function fixed.
  - 1.09 Keyfile-Support.
  - 1.10 Many new File types.
  - 1.11 During  
    ShowDetect  
        and  
    SetComment  
        AView is using the pre  
        defined file types.
  - 1.12 AView allows to remove the "" around the file names. (Requested by Steffen Koch)
  - 1.13 A new  
    %a  
        option, allows to specify a argument.
  - 1.14 An other manual update.
  - 1.15 WB start is working correct now.
-

- 1.16 Enlarged  
       ASCII  
       command to support ESC sequences.
- 1.17 The whole detecting file detecting procedure enhanced.
- 1.17 New AViewPrefs. No longer Up/Down gadgets. A new  
       Edit  
       | window implemented.
- 1.29 New  
       expert mode  
       .
- 1.30 Small bug fixes and an other manual update.
- 1.31 Menus added.  
       AViewPrefs-preferences  
       requester added. Currently there is only one  
       point: Expert mode.
- 1.32 AView adjustment to support the new features.
- 1.33 Small bug fix. (  
       Menu open  
       )
- 1.34 Screen title is viewing register information.
- 1.35 = 1.34 (small catalog changes)

## 1.13 Thanx

Thanx  
 -----

Timo Hegemann : Beta and Enforcer testing

Andreas Vierkant : Beta testing

Jerome Chesnot : for the france translation.

Andreas Boerner : for tips, suggestions and bug reports.

Steffen Koch : for his suggestions.

And of course many thanks to all registered users.

## 1.14 Buglist

Buglist  
 -----

\textdegree{} Someone told me that AViewPrefs is crashing on an 68060. I am not able  
 to detect the problem. If you are using an 68060, then send me your test  
 results. ↔

\textdegree{} Currently there are no other known bugs.

## 1.15 The Author

The Author  
-----

If you find some bugs in AView, please send me a message, so that I can correct these bugs in the next version.

Guido Mersmann  
Glatzer Straße 12  
48477 Hörstel  
Deutschland

FIDONET : 2:2449/246.15  
INTERNET: geit@studST.FH-Muenster.DE

Someone, who don't want to send me the money via post, but remit it, can do this as well:

Sparkasse Ibbenbüren

Bankleitzahl: 403 510 60  
Kontonummer : 71107791

New versions and updates are available at:

Aminet: "util/misc/aview.lha"  
SupportBox: The Kiteman 2:2449/246 TEL: (+49) 05971-15835  
"aview.lha"

Use the Magic "Amiga" to receive a full list of available Software for your Amiga.

## 1.16 Other programs !

Other programs!  
-----

BoulderDäsh [game/jump/boulderdaesh.lha]  
-----

Clone of the original Boulderdäsh for Amiga. It is the only version running faster on Amiga than the original. It looks and behaves like the original. I was asked, where the C64 is. (-8 And all this on a 68000 without Fastmem.

SimpleCat [dev/misc/simplecat.lha]  
-----

This tool is for programmers and user. Using this program you can change catalogues of programs with your favourite text editor in the easiest way. These programs must include a CS file. Search for it in the archive of the catalogues drawer.

---

Afind [util/misc/afind.lha]  
-----

With this program you can search for programs on your Aminet CDs in the easiest way. It is the same than the original tool on the CD, but here you can use AminetCDs and AminetSets mixed. So you are not forced to search for a program on the first 5 Aminet CDs, that is on AminetSet1 Disk b.

Guideformat [text/edit/guideformat.lha]  
-----

GuideFormat is a simple tool. It is easy to format text blocks, even if there are "links" or other guide commands in.

SiedlerBoot [game/patch/siedlerboot.lha]  
-----

This allows to start the Settlers direct via CD and save the scores on HD. There is no hard disk install required. SiedlerBoot requires the "Amiga Plus Sonderheft 9" cover CDROM.

## 1.17 Technical information about AView

Technical information about AView  
-----

AView is 100% assembler coded.

The sourcecode takes 234320 bytes in 34 files. AView is assembled with DevPac 3.04. DevPac needs 4 seconds and 755150 KB each time.

AView was programmed on an A2000-030/25Mhz (9 MEG).

I wrote this guide using GoldEd (3.x/4.x) and his fantastic fold features together with "GuideFormat" to format lines containing links.

## 1.18 The File Identification Commands

The File Identification Commands  
-----

The identifier gadget allows to define own file types, by using the following commands.

GENERAL  
-----

\textdegree{} All commands must be separated by one space as minimum.

---

```
\textdegree{} Command names are case insensitiv. "
      MATCH
      " = "MaTcH"
```

```
\textdegree{} All commands are only scanning the fist 8 KB buffer of an file. ←
      If a
      command is reading beyond this 8KB the hole file type fails. A file
      size which is smaller than 8KB will be handled like a smaller buffer.
```

#### A List Of All Commands

-----

|          |                                  |
|----------|----------------------------------|
| MATCH    | Check specified Bytes or Strings |
| FIND     | Search for Bytes or Strings      |
| NAME     | Check file name                  |
| FILESIZE | Check file size.                 |
| ASCII    | Check for ASCII file             |
| AND      | AND-operation                    |
| OR       | OR-operation                     |
| NOT      | NOT-operation                    |
| ()       | Bracket operation                |

## 1.19 Command: MATCH

Match  
-----

TEMPLATE:

MATCH [Offset/N,] String

SHORT:

---



This command allows to check bytes or strings starting on an direct offset.

EXAMPLES:

```
MATCH "Hello"
```

Checks for an "Hello" in the first 5 bytes of an file.

```
MATCH 1,"Hallo"
```

Checks for an "Hello". This time the offset is one.

```
MATCH 27,$1234
```

This match is looking for an \$1234 on file offset 27.

```
MATCH 0,"FORM????ILBM"
```

This detects a "FORM" on offset zero and an "ILBM" on offset eight. An "?" means no further check. This Byte will be skipped.

IMPORTANT:

The offset is optional and always zero if not specified.

The offset is always decimal.

The String must start with an " or an \$!

The String may contain "?". This bytes won't be checked.

## 1.20 Command: FIND

```
FIND
```

TEMPLATE:

```
FIND [Start/N,] [End/N,] String
```

SHORT:

This command allows to search for an string between two offsets.

EXAMPLES:

```
FIND "Hello"
```

AView is searching the whole buffer! The search pattern is "Hello".

---

```
FIND 10,"Smurf"
```

AView is searching the string "Smurf" between 0 and 10.

```
FIND 1,10,"Smurf"
```

AView is searching for the string "Smurf" between 1 and 10. 10 is the last offset. This means AView will search the last time by using offset 10: "S" on offset 10, "m" on offset 11 and so on.

```
FIND 1,127,$0001
```

AView is searching for the bytes \$00 and \$01 between "1" and "127".

IMPORTANT:

The offsets are optional. If only one is specified this is the search end. If nothing is declared the whole buffer will be searched.

Both offsets are decimal.

The String must start with an " or an \$!

## 1.21 Command: NAME

```
NAME
```

TEMPLATE:

```
NAME Pattern/N
```

SHORT:

This command compares the file name by using dos pattern.

EXAMPLES:

```
NAME "#?.gif"
```

This command will be only successful on files like "Smurf.gif"!

```
NAME "#?.(jpeg|jpg)"
```

Only file names with .jpeg or .jpg at the end are valid.

IMPORTANT:

---

The pattern must be enclosed with "!

## 1.22 Command: FILESIZE

FILESIZE

-----

TEMPLATE:

FILESIZE Number/N

SHORT:

This command compares the file size and the declared number.

EXAMPLES

FILESIZE 901120

This command only comes true if the file size is equal to the specified number.

FILESIZE 1024

AND

NAME

"BootBlock#?"

This file size must be 1024 and the file name must start with "BootBlock".

IMPORTANT:

The number must be decimal.

## 1.23 Command ASCII

ASCII

-----

TEMPLATE:

ASCII

SHORT:

This command checks for non ASCII Bytes and will return "1" on the condition that there is non.

EXAMPLES:

ASCII

## 1.24 Command: AND

AND

-----

TEMPLATE:

AND

SHORT:

This command allows to connect  
 commands  
 via an AND operation.

EXAMPLES:

NAME

"#?.txt" AND

ASCII

This will be successful if the name ends with an ".txt" and it ←  
 is an

ASCII

file.

NAME

"#?.txt" &

ASCII

This is equal to the last example.

NAME

"#?.guide" &

ASCII

AND

FIND

40, "@Database"

The file name must end with an ".guide" and the file must contain an "@Database" starting within the first 40 bytes.

IMPORTANT:

You are allowed to use the short version of AND: "&"

An AND operation is always solved before an  
Or  
operation. It is  
possible to override this by using Brackets

## 1.25 Command: OR

OR

----

TEMPLATE:

OR

SHORT:

This command allows to connect  
single commands  
with an OR operation.

EXAMPLES:

```
NAME
"#?.txt" OR
ASCII
The
NAME
must end with ".txt" or the file must be an
ASCII
```

```
NAME
"#?.txt" |
ASCII
This example is equal to the first one.
```

```
NAME
"#?.guide" |
ASCII
OR
FIND
40,"@Database"
```

One of the three commands must succeed to make this file identification successful.

---

IMPORTANT:

"|" is the short version of "OR"

An

AND operation has an higher priority. The only way to change this priority is to use Brackets.

## 1.26 Command: NOT

NOT

-----

TEMPLATE:

NOT

SHORT:

This command causes a negation.

EXAMPLES:

```
NAME
  "#?.hft"
AND
  NOT
ASCII
```

This line will check for an file that is not ASCII. Additional to that the file name must end with an ".hft".

```
NAME
  "#?.hft"
AND
  ~
ASCII
```

This example is equal to the first one.

~ (

```
NAME
  "#?.data" &
ASCII
)
```

No ".data" and no

```
ASCII
file.
```

IMPORTANT:

"~" is the short version of "NOT"

A NOT operations is always preferred. The only exception are brackets

=> "NOT NAME "Hello" AND ASCII" = "(NOT NAME "Hello) AND ASCII"

## 1.27 COMMAND: ()

( and )

-----  
TEMPLATE:

```
(
)
```

SHORT:

Brackets allow to prefer operations.

EXAMPLES:

```
NAME
"#?.guide"
AND
(
ASCII

OR

FIND
40,"@Database")
```

File name must end with ".guide". Also an ASCII or an "@Database" within the first 40 bytes is required, to succeed this file type.

This example is the equal to "a\*(b+b)"

IMPORTANT:

\textdegree{} Make sure that all brackets are closed. A missing bracket causes ↔ the

file type to fail.

\textdegree{} File types are always solved inside out!

## 1.28 Pattern: Name/M

Name/M  
-----

"Name" defines the name of the file you want to view. The "/M" means that AView is able to process as much filenames as possible. Every file name is able to handle DOS pattern matching : E.G "#?.gif"

Examples:

AView #?.gif

AView Smurf.jpg PapaSchlumpf.gif #?smurf#?

You are allowed to specify an path as well:

AView sys:pictures/smurfs/Papa.gif sys:Pictures/old/A500.pic

There are no limits.

## 1.29 Pattern: All/S

All/S  
-----

"All" allows to scan all sub directories.

## 1.30 Pattern: ARG=To=From/K

ARG=To=From/K  
-----

"ARG", "To" or "From" are very useful. This element will replace the

%a  
string.

For example "lha":

"LHA x -M -X %n %a"

%n  
will be replaced by the name specified with the



```

name
  pattern.
  %a
    is normally replaced by "t:". This means AView is. ←
    decrunching an lha
file to "T:".

```

You are able to specify the default %a argument separately for each file type.

So "AView Dummy.lha ARG ram:" will decrunch the file "Dummy.lha" direct to "ram:".

Important: To use this %a mode one of the following key words is required: ARG, To or From

Examples:

AView BoulderDäsh.lha to ram:

This example will decrunch the archive "BoulderDäsh.lha" to "ram:"

AView BoulderDäsh.lha from ram:

The result of this example is exact like the first. It looks stupid, because we decrunch to "RAM:" and not from!

### 1.31 Pattern: SetComment=SC/S

```

SetComment=SC/S
-----

```

"SetComment" or "SC" is switching AView into the marking mode. All files will be scanned and the result will be written into the file comment.

AView is not able to start any commands during this mode!

### 1.32 Pattern: ShowDetect=SD/S

```

ShowDetect=SD/S
-----

```

"ShowDetect" or "SD" is viewing the file type of any file.

AView is not able to start any commands during this mode!

---

### 1.33 Pattern: Debug/S

Debug/S  
-----

"Debug" shows any problems during file identification. This switch is not useful for normal users. At this moment AView is only able to view unknown commands and syntax errors.

### 1.34 The AView preferences

The AView Preferences  
-----

The AView preferences program allows to modify all file types and the action behind. AViewPrefs supports

menu  
functions as well.

Sort by

Name

Select

Delete

All

New

Save

Use

Abort

### 1.35 Gadget: Sort by

Sort by  
-----

---

This Gadget is only available during  
expert mode  
This gadget allows to sort all entries by name or sub class. ↔  
Under normal  
conditions all entries are sorted by name.

AView is processing all entries by  
sub class  
.

If you have created own file types, sorting all entries by  
sub class  
is useful to find problems.

### 1.36 Gadget: Name

Name  
-----

This list contains all file types AView is able to detect. Use  
select  
and if needed  
New  
to define new entries.

You are able to scroll via cursor keys. Return or double clicking on  
one entry opens the  
edit  
window.

### 1.37 Gadget: Select

Select  
-----

This gadget opens a select window. It allows to select one of the pre  
defined identifier.

### 1.38 Gadget: Delete

Delete  
-----

This gadget deletes the actual identifier.

---

### 1.39 Gadget: All

All  
-----

All pre defined file types will be added to your prefs list.

Already existing entries won't be modified.

### 1.40 Gadget: New

New  
-----

This Gadget is only available during  
expert mode

You are able to define complete new entries (file types). Just ↔  
enter a  
name and AViewPrefs will add your entry to the list.

### 1.41 Gadget: Save

Save  
-----

This adjustments will be saved and the requester will be existed.

### 1.42 Gadget: Use

Use  
-----

All adjustments will be used will be existed.

### 1.43 Gadget: Abort

Abort  
-----

Cancels all adjustments and quit requester.

### 1.44 The Edit Window

---

### The Edit Window

---

This window allows all adjustments of the file types.

Name

Class

Sub Class

Exec type

Stack

Command

Current Type

Current Directory

WB Output File

CLI Output File

Identifier

Default %a

## 1.45 Gadget: Name

Name

---

This defines the name of this file type. Do not define a name twice, because this could cause problems with later versions.

## 1.46 Gadget: Class

---

### Class

-----

This Gadget is only available during  
expert mode

This Gadget allows to specify the type of the current file type ↔  
. If you  
don't find a match you should use the type "User".

At the moment this class is used for output and sorting only. Please  
define this class correct to be on the save side.

## 1.47 Gadget: Sub Class

### Sub Class

-----

This Gadget is only available during  
expert mode

This is the most important gadget. It specifies where to ↔  
insert this  
file type.

A file detected by name only is not as save as a detection by contents.

This means a file detected by name only should be checked at the end of  
the list. Just think about an lha file that is named "archive.c". This  
file may be interpreted as an c source if the sequence of an file  
identifier is not correct.

I defined some classes my name:

|                 |                                                                                                                                                                         |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| User:           | This class is used for user defined file types. You<br>should use one of the other types.                                                                               |
| Normal:         | Under normal cases this is the best selection for<br>new cases.                                                                                                         |
| Medium:         | This class is used for groups, (Kickstart - global,<br>Icon - global) because it doesn't make any sense<br>to check a single Kickstart after a global definition.       |
| xxx or name:    | e.g. "(FIND "<html>" OR NAME "#?.htm(l )") AND ASCII"<br>This is better than "Name and ASCII", but not as good<br>as "Medium", because a file may not contain "<html>"! |
| Name and ASCII: | This is reserved for file types that are only<br>identified by name and ASCII.                                                                                          |
| Name only:      | Checking a file by using only a name is one of the<br>worst methods, but sometimes there is no other<br>choice. So this is reserved for name only file<br>types.        |

Do not use the other classes, if not necessary. If you create file  
types missing in AView defaults, then please send me your the prefs  
file.

By the way: It is possible to use the menu

```

        open
        function to load the
    "Data/AView.data" file. This file contains all pre defined file types.
    Save this data by using the menu
        Save as
        .

```

Be careful a wrong modification and AView isn't working correct. Create a backup of this file before changing the files contents.

## 1.48 link

Exec Type

-----

This Gadget allows to specify the way AView is starting the command

Under normal conditions this is the CLI mode, but some programs may depends on an start via Icon.

## 1.49 Gadget: Stack

Stack

-----

Each

command

needs an amount of stack memory. Normally 8192 Bytes should be enough. If the program crashes increase this value.

## 1.50 Gadget: Command

Command

-----

If AView detects a known file type, it will start this command. You should take a look on the following % commands.

%p : includes the path of the file specified by  
 name  
 .(e.g "SYS:Pictures/")

%~p : like %p! No "" this time.

%n : includes the file name of the file specified by  
 name  
 .  
 (e.g "smurfs.gif")

%~n : like %n! No "" this time.

%f : The complete file name of the file will be included.  
(e.g. "SYS:Pictures/Smurf.gif")

%~f : like %f! No "" this time.

%a : This will be replaced by an  
Arg  
. The default  
value may be modified by using the  
Default %a  
Gadget.

%~a : like %f! No "" this time.

There are only four command patterns (p,n,f,a). If you include an ~ after the % this means 'insert no ""'

If this command is processed during WB mode all % patterns will be removed. The only exclusion is the %a (%~a) pattern.

If you need a "%" in your command then use "%%! AView replaces them to one "%!"

## 1.51 Gadget: Current Type

Current Type

-----  
This Gadget allows to specify a dos command feature that is called "Current Dir". It works like the DOS-CD command.

File directory : Makes a CD to the path of the identified file.  
Current : No Change!  
Specified Dir : The Path is defined by the  
CurrentDir

## 1.52 Gadget: Current Directory

Current Directory

-----  
This Gadget is only available if the  
Current Type  
gadget is switched  
to "specified directory"!

## 1.53 Gadget: WB Output File



#### WB Output File

---

Any Output of the  
 Command  
 will be directed to the file specified by  
 this gadget. This redirection is only valid if AView is running via  
 workbench. (Double click on AView icon or Toolmanager is starting AView  
 in WB Mode)

### 1.54 Gadget: CLI Output File

#### CLI Output File

---

Any Output of the  
 Command  
 will be directed to the file specified by  
 this gadget. This redirection is only valid if AView is running via  
 CLI.

### 1.55 Gadget: Identifier

#### Identifier

---

This Gadget is only available during  
 expert mode  
 It a very important gadget. It is the heart of the ↔  
 AView file  
 recognition system. This is done by some  
 commands  
 .

### 1.56 Gadget: Default %a

#### Default %a

---

This is the default for the  
 %a  
 pattern of the  
 command  
 gadget. The  
 contents of this gadget will be used, if no other argument is specified  
 by the  
 AView Dos Command

---

## 1.57 The File Type Selection

The File Type Selection

---

This window is viewing all default file types of  
AView\_Prefs  
.

This list is sorted by file classes, so it is easy to configurate  
AView.

Use the Cursor keys to select an entry and select via return key. You  
are also able to use the mouse and double click to select.

## 1.58 The Settings of AViewPrefs

The Settings of AViewPrefs

---

It is easier than it sounds. This window allows to modify some  
AViewPrefs internal Features. These settings have no effect on AView.

The expert mode

Save

Use

Abort

## 1.59 Gadget: Expert Mode

Expert Mode

---

This gadget should be used by expert users only. It enables an special  
expert mode. This mode enables hidden gadgets to make additional  
adjustments to your configuration.

You should understand the priority controlled file recognizing system  
and it is important that you understand the main problem behind.

For me the author it isn't easy to find error in the pre defined file  
types. This means it won't be easy for you.

---

## 1.60 Gadget: Save

Save

-----

All settings made in this window will be saved and this requester will be closed.

## 1.61 Gadget: Use

Use

-----

The settings will be used.

## 1.62 Gadget: Abort

Abort

-----

The old settings will be restored.

## 1.63 The menus of AViewPrefs

The menus of AViewPrefs

-----

These are the menu items of AViewPrefs:

Project

Open...

SaveAs...

Quit

Default

Reset to Default

Last Saved

Restore

Preferences

AView Prefs

## 1.64 Menu: Open

Open  
-----

This item allows to load a prefs file. The current settings will be lost!

## 1.65 Menu: SaveAs

SaveAs  
-----

This item allows to save the prefs file.

## 1.66 Menu: Quit

Quit  
-----

Quit AView prefs and cancel all modifications.

## 1.67 Menu: Reset to Default

Reset To Default  
-----

The file type list will be cleared a set to the default entries.

## 1.68 Menu: Last Saved

Last Saved  
-----

This menu point forces an reload of the last saved preferences.

---

## 1.69 Menu: Restore

Restore  
-----

This menu point forces an reload of the last used preferences.

## 1.70 Menu: AView Prefs

AView Prefs  
-----

This menu will open the  
AViewPrefs-preferences window  
.

## 1.71 Frequently ask Questions

Frequently ask Questions  
-----

AView and AViewPrefs doesn't run!

The online help isn't working! Why?

The online help is english only.

AViewPrefs sometimes is very slow.

## 1.72 AView and AViewPrefs aren't running

AView and AViewPRefs aren't running  
-----

The OS2.1+ or OS3.0+ version of AView require special catalog files. These files contain all text information these programs need during operation. You will find the english catalog files in "AView/Catalogs/english/AView.catalog".

It is not allowed to move a AView. This means after installing all file must stay on there place. This is needed, because AView and AViewPrefs must be able to find special files like "Data/AView.Data", "Prefs/AView.prefs" and so on.

---

### 1.73 The online help isn't working! Why?

The online help isn't working! Why?

-----

AView is using the "locale.library" and the locale settings. If the system is running in english you need the english guide file.

In addition to that the "AmigaGuide.library" is required.

### 1.74 The online help is english only.

\textdegree{} The online help is english only.

If you are using OS2.0 you should copy your german guide file from "docs/deutsch" to "docs/english":

```
CD sys:tools/AView/Docs
copy deutsch/aview.guide english/
```

There is no "deutsch/Avview.guide"? Then you disabled the german documentation during AView installation. The easiest solution is to install AView again. (AView.install recognises the pre install and is switching to an update mode)

### 1.75 AViewPrefs sometimes is very slow.

AViewPrefs sometimes is very slow.

-----

This is normal. AViewPrefs is sorting his data file and your preferences by name. The Sorting procedure takes some seconds.

During the save operation AView is sorting you preference by

```
sub class
    . After saving all entries are resorted by the mode specified
in the
    sort by
    gadget. This is required because AView needs the
settings sorted by
    sub class
    to process them correct. This procedure is
better than let AView sort every time.
```

You are able to test the sorting speed by using the expert mode. This will activate the

```
sort by
    gadget. Use the
all
    gadget to see how fast
is you computer. (-8
```