

**Filer**

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

	<i>TITLE :</i>  Filer		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 22, 2024	

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

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Filer</b>	<b>1</b>
1.1	Filer.guide . . . . .	1
1.2	Filer.guide/BI_CRIGHT . . . . .	2
1.3	Filer.guide/BI_REG . . . . .	2
1.4	Filer.guide/IN_PURPOSE . . . . .	4
1.5	Filer.guide/IN_SYSR . . . . .	4
1.6	Filer.guide/IN_INST . . . . .	5
1.7	Filer.guide/UF_MW . . . . .	5
1.8	Filer.guide/UF_MW_SBAR . . . . .	5
1.9	Filer.guide/UF_MW_LV . . . . .	6
1.10	Filer.guide/UF_MW_DLV . . . . .	6
1.11	Filer.guide/UF_MW_PATH . . . . .	7
1.12	Filer.guide/UF_MW_COPY . . . . .	7
1.13	Filer.guide/UF_MW_CD . . . . .	7
1.14	Filer.guide/UF_MW_PAR . . . . .	7
1.15	Filer.guide/UF_MW_CACHE . . . . .	8
1.16	Filer.guide/UF_MW_PAT . . . . .	8
1.17	Filer.guide/UF_MW_ALL . . . . .	8
1.18	Filer.guide/UF_MW_NONE . . . . .	9
1.19	Filer.guide/UF_MW_PANEL . . . . .	9
1.20	Filer.guide/UF_MENU . . . . .	9
1.21	Filer.guide/UF_REQ . . . . .	13
1.22	Filer.guide/UF_FN . . . . .	14
1.23	Filer.guide/UF_FN_COPY . . . . .	14
1.24	Filer.guide/UF_FN_COPYAS . . . . .	15
1.25	Filer.guide/UF_FN_DELETE . . . . .	15
1.26	Filer.guide/UF_FN_DU . . . . .	16
1.27	Filer.guide/UF_FN_DUPLICATE . . . . .	16
1.28	Filer.guide/UF_FN_FILENOTE . . . . .	16
1.29	Filer.guide/UF_FN_MAKEDIR . . . . .	16

1.30 Filer.guide/UF_FN_MAKELINK . . . . .	17
1.31 Filer.guide/UF_FN_MOVE . . . . .	17
1.32 Filer.guide/UF_FN_MOVEAS . . . . .	17
1.33 Filer.guide/UF_FN_PROTECT . . . . .	18
1.34 Filer.guide/UF_FN_RELABEL . . . . .	18
1.35 Filer.guide/UF_FN_RENAME . . . . .	18
1.36 Filer.guide/UF_FN_CHOWN . . . . .	19
1.37 Filer.guide/UF_FN_CHGRP . . . . .	19
1.38 Filer.guide/UF_FN_ACTION . . . . .	19
1.39 Filer.guide/UF_FN_HEXDUMP . . . . .	20
1.40 Filer.guide/UF_FN_RUN . . . . .	20
1.41 Filer.guide/UF_FN_TYPE . . . . .	20
1.42 Filer.guide/UF_FN_ADD . . . . .	21
1.43 Filer.guide/UF_FN_ADDDEL . . . . .	21
1.44 Filer.guide/UF_FN_EXTRACT . . . . .	22
1.45 Filer.guide/UF_FN_LIST . . . . .	22
1.46 Filer.guide/UF_FN_EDIT . . . . .	22
1.47 Filer.guide/UF_FN_FORMAT . . . . .	23
1.48 Filer.guide/UF_FN_SEARCH . . . . .	23
1.49 Filer.guide/UF_FN_QUIT . . . . .	24
1.50 Filer.guide/UF_KBD . . . . .	24
1.51 Filer.guide/UF_KBD_BASIC . . . . .	24
1.52 Filer.guide/UF_KBD_CURSOR . . . . .	25
1.53 Filer.guide/UF_KBD_LV . . . . .	25
1.54 Filer.guide/UF_KBD_CMD . . . . .	26
1.55 Filer.guide/UF_KBD_DIR . . . . .	27
1.56 Filer.guide/UF_ACT . . . . .	28
1.57 Filer.guide/UF_TEXT . . . . .	28
1.58 Filer.guide/UF_CF . . . . .	29
1.59 Filer.guide/UF_CF_DISPLAY . . . . .	31
1.60 Filer.guide/UF_CF_SCREEN . . . . .	31
1.61 Filer.guide/UF_CF_WINDOW . . . . .	32
1.62 Filer.guide/UF_CF_COLOR . . . . .	32
1.63 Filer.guide/UF_CF_RGB . . . . .	33
1.64 Filer.guide/UF_CF_FONT . . . . .	33
1.65 Filer.guide/UF_CF_PANELFONT . . . . .	34
1.66 Filer.guide/UF_CF_SCREENFONT . . . . .	34
1.67 Filer.guide/UF_CF_BUTTON . . . . .	35
1.68 Filer.guide/UF_CF_XBUTTON . . . . .	35

---

1.69 Filer.guide/UF_CF_KEY . . . . .	36
1.70 Filer.guide/UF_CF_CLOSEGADGET . . . . .	36
1.71 Filer.guide/UF_CF_STARTUP . . . . .	37
1.72 Filer.guide/UF_CF_SHUTDOWN . . . . .	37
1.73 Filer.guide/UF_CF_SET . . . . .	38
1.74 Filer.guide/UF_CF_COPYBUFFER . . . . .	42
1.75 Filer.guide/UF_CF_FORMAT . . . . .	42
1.76 Filer.guide/UF_CF_HIDE . . . . .	43
1.77 Filer.guide/UF_CF_HISTORY . . . . .	44
1.78 Filer.guide/UF_CF_VIEWER . . . . .	44
1.79 Filer.guide/UF_CF_VIEWERDISPLAY . . . . .	45
1.80 Filer.guide/UF_CF_VIEWERFONT . . . . .	45
1.81 Filer.guide/UF_CF_ADDPATH . . . . .	45
1.82 Filer.guide/UF_CF_STACK . . . . .	46
1.83 Filer.guide/UF_CF_CONHANDLER . . . . .	46
1.84 Filer.guide/UF_CF_EDITOR . . . . .	47
1.85 Filer.guide/UF_CF_ARCDEF . . . . .	47
1.86 Filer.guide/UF_CF_DEFARCEXT . . . . .	48
1.87 Filer.guide/UF_CF_IFFDEF . . . . .	49
1.88 Filer.guide/UF_CF_TOOL . . . . .	49
1.89 Filer.guide/UF_CF_REXXTOOL . . . . .	50
1.90 Filer.guide/UF_CF_CLASS . . . . .	50
1.91 Filer.guide/UF_CF_REXXCLASS . . . . .	51
1.92 Filer.guide/UF_RX . . . . .	51
1.93 Filer.guide/UF_RX_VERSION . . . . .	54
1.94 Filer.guide/UF_RX_EXIT . . . . .	54
1.95 Filer.guide/UF_RX_FILERTOFRONT . . . . .	55
1.96 Filer.guide/UF_RX_CD . . . . .	55
1.97 Filer.guide/UF_RX_GETCD . . . . .	55
1.98 Filer.guide/UF_RX_ACTION . . . . .	56
1.99 Filer.guide/UF_RX_ADD . . . . .	56
1.100Filer.guide/UF_RX_CHGRP . . . . .	57
1.101Filer.guide/UF_RX_CHOWN . . . . .	57
1.102Filer.guide/UF_RX_COPY . . . . .	57
1.103Filer.guide/UF_RX_DELETE . . . . .	58
1.104Filer.guide/UF_RX_DU . . . . .	58
1.105Filer.guide/UF_RX_DUPLICATE . . . . .	58
1.106Filer.guide/UF_RX_EDIT . . . . .	59
1.107Filer.guide/UF_RX_EXTRACT . . . . .	59

---

1.108Filer.guide/UF_RX_FILENOTE . . . . .	59
1.109Filer.guide/UF_RX_FORMAT . . . . .	60
1.110Filer.guide/UF_RX_HEXDUMP . . . . .	60
1.111Filer.guide/UF_RX_LIST . . . . .	60
1.112Filer.guide/UF_RX_MAKEDIR . . . . .	61
1.113Filer.guide/UF_RX_MAKELINK . . . . .	61
1.114Filer.guide/UF_RX_MOVE . . . . .	62
1.115Filer.guide/UF_RX_PROTECT . . . . .	62
1.116Filer.guide/UF_RX_QUIT . . . . .	62
1.117Filer.guide/UF_RX_RELABEL . . . . .	63
1.118Filer.guide/UF_RX_RENAME . . . . .	63
1.119Filer.guide/UF_RX_RUN . . . . .	63
1.120Filer.guide/UF_RX_SEARCH . . . . .	64
1.121Filer.guide/UF_RX_TYPE . . . . .	64
1.122Filer.guide/UF_RX_ALL . . . . .	64
1.123Filer.guide/UF_RX_BANK . . . . .	65
1.124Filer.guide/UF_RX_ICONIFY . . . . .	65
1.125Filer.guide/UF_RX_NONE . . . . .	66
1.126Filer.guide/UF_RX_PATTERN . . . . .	66
1.127Filer.guide/UF_RX_SELECTCACHE . . . . .	67
1.128Filer.guide/UF_RX_READSOURCEDIR . . . . .	67
1.129Filer.guide/UF_RX_READTARGETDIR . . . . .	68
1.130Filer.guide/UF_RX_READLEFTDIR . . . . .	68
1.131Filer.guide/UF_RX_READRIGHTDIR . . . . .	69
1.132Filer.guide/UF_RX_ENTERDIR . . . . .	69
1.133Filer.guide/UF_RX_OTHERDIR . . . . .	70
1.134Filer.guide/UF_RX_PARENTDIR . . . . .	70
1.135Filer.guide/UF_RX_UP . . . . .	71
1.136Filer.guide/UF_RX_PAGEUP . . . . .	71
1.137Filer.guide/UF_RX_HOME . . . . .	72
1.138Filer.guide/UF_RX_DOWN . . . . .	72
1.139Filer.guide/UF_RX_PAGEDOWN . . . . .	73
1.140Filer.guide/UF_RX_END . . . . .	73
1.141Filer.guide/UF_RX_LEFT . . . . .	74
1.142Filer.guide/UF_RX_PAGELEFT . . . . .	74
1.143Filer.guide/UF_RX_LEFTMOST . . . . .	74
1.144Filer.guide/UF_RX_RIGHT . . . . .	75
1.145Filer.guide/UF_RX_PAGERIGHT . . . . .	75
1.146Filer.guide/UF_RX_RIGHTMOST . . . . .	76

---

1.147Filer.guide/UF_RX_CURSORUP . . . . .	76
1.148Filer.guide/UF_RX_CURSORPAGEUP . . . . .	77
1.149Filer.guide/UF_RX_CURSORHOME . . . . .	77
1.150Filer.guide/UF_RX_CURSORDOWN . . . . .	78
1.151Filer.guide/UF_RX_CURSORPAGEDOWN . . . . .	78
1.152Filer.guide/UF_RX_CURSOREND . . . . .	79
1.153Filer.guide/UF_RX_CURSORSELECT . . . . .	79
1.154Filer.guide/UF_RX_CURSORSELECTDOWN . . . . .	80
1.155Filer.guide/UF_RX_CURSORSELECTUP . . . . .	80
1.156Filer.guide/UF_RX_CURSORACTION . . . . .	81
1.157Filer.guide/UF_RX_LOCKFILER . . . . .	81
1.158Filer.guide/UF_RX_UNLOCKFILER . . . . .	82
1.159Filer.guide/UF_RX_PANEL . . . . .	82
1.160Filer.guide/UF_RX_STATUS . . . . .	83
1.161Filer.guide/UF_RX_HISTORY . . . . .	83
1.162Filer.guide/UF_RX_CLEARHISTORY . . . . .	83
1.163Filer.guide/UF_RX_RX . . . . .	84
1.164Filer.guide/UF_RX_TEMPFILENAME . . . . .	84
1.165Filer.guide/UF_RX_ALERTBOX . . . . .	85
1.166Filer.guide/UF_RX_QUESTBOX . . . . .	85
1.167Filer.guide/UF_RX_FILERREQ . . . . .	86
1.168Filer.guide/UF_RX_GETSTRING . . . . .	86
1.169Filer.guide/UF_RX_SETSTRING . . . . .	87
1.170Filer.guide/UF_RX_EXEC . . . . .	87
1.171Filer.guide/UF_RX_EXECANDWAIT . . . . .	88
1.172Filer.guide/UF_RX_FOREACH . . . . .	88
1.173Filer.guide/UF_RX_GETSCREENNAME . . . . .	89
1.174Filer.guide/UF_RX_GETSOURCEPATH . . . . .	89
1.175Filer.guide/UF_RX_GETTARGETPATH . . . . .	89
1.176Filer.guide/UF_RX_GETLANGUAGE . . . . .	90
1.177Filer.guide/UF_RX_GETNUMENTRIES . . . . .	90
1.178Filer.guide/UF_RX_GETNAME . . . . .	91
1.179Filer.guide/UF_RX_GETINFO . . . . .	91
1.180Filer.guide/UF_RX_SELECTENTRY . . . . .	92
1.181Filer.guide/UF_RX_TOGGLEENTRY . . . . .	92
1.182Filer.guide/UF_RX_UPDATESOURCEDIR . . . . .	93
1.183Filer.guide/UF_RX_UPDATETARGETDIR . . . . .	93
1.184Filer.guide/UF_EXT . . . . .	93
1.185Filer.guide/UF_CX . . . . .	94

---

1.186Filer.guide/UF_CX_ARGS . . . . .	94
1.187Filer.guide/UF_CX_HOTKEY . . . . .	95
1.188Filer.guide/UF_CX_APPICON . . . . .	96
1.189Filer.guide/OT_UPD . . . . .	96
1.190Filer.guide/OT_SUP . . . . .	97
1.191Filer.guide/OT_HIST . . . . .	97
1.192Filer.guide/OT_FUT . . . . .	100
1.193Filer.guide/OT_ACK . . . . .	100



# Chapter 1

## Filer

### 1.1 Filer.guide

Filer 3.23

An Intuition controlled file manager

Copyright (c) 1989-95 Matthias Scheler

- ShareWare -

Chapters for all users...

Copyright	Your rights.
Registration	How to become a registered user.

Introduction...

Purpose	What is Filer made for?
System requirements	Which computer can run Filer?
Installation	How to install Filer.

Using Filer...

Main window	Files, Directories, etc.
Menus	Pull-Down Menus of the Main window.
Standard requesters	Filer's standard requesters.
Builtin functions	Copying, deleting, etc.
Keyboard control	How to control Filer via keyboard.
Auto Action	Automatic actions on Files.
Text viewer	Filer's internal Text viewer
Configuration file	How to configure Filer.
ARexx interface	Filer's ARexx Interface.
External programs	Starting external programs.
FilerCom	Filer's Commodity.

Other topics...

Updates	How to get updates.
Support	How to reach the author.

---

History	History of Filer.
Future	Future of Filer.
Acknowledgements	The author wishes to thank...

## 1.2 Filer.guide/BI\_CRIGHT

Basic Informations

\*\*\*\*\*

Copyright and other legal stuff

=====

Copyright (c) 1989-95 Matthias Scheler  
Schützenstraße 18  
33178 Borchten  
FR Germany

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

No guarantee of any kind is given that the programs described in this document are 100% reliable. You are using this material at your own risk. The author can not be made responsible for any damage which is caused by using these programs.

Permission is granted to include this package in Public-Domain collections, especially in Fred Fish's Amiga Disk Library (including CD-ROM versions of it). The distribution file may be uploaded to Bulletin Board Systems or FTP servers. If you want to distribute this program you must use the original distribution archive Filer3\_23.lha.

None of the programs may be included or used in commercial programs unless by written permission from the author.

None of the programs may be modified in any way. This especially includes changing the copyright notices or removing any of the ShareWare restrictions.

None of the programs may be used on any machine which is used for the research, development, construction, testing or production of weapons or other military applications. This also includes any machine which is used for training persons for any of the above mentioned purposes.

## 1.3 Filer.guide/BI\_REG

Registration

=====

As you may have noticed in the copyright I'm working since four

---

years at Filer. It always consumed and will continue to consume a large amount of my time.

I cannot afford just working for fun. Thus, I decided to release Filer as shareware. I already tried to release Filer as a commercial product but the german software company which licensed it was not able to release it for over one year. So I cancelled the contract.

The unregistered version of Filer pops up the About requester regularly to remind you to register and has some features disabled:

- directory cache inactive
- can't format disks
- command panel's bank can't be switched

Registered users will be shipped a disk with the newest public release of Filer, along with a personalized, so-called "keyfile". It enables all the missing features and disables the ShareWare reminders. This keyfile will work with all future releases of Filer, so you can simply download the latest version from your local bulletin board without having to wait weeks for your update passing through the slow mail channels. The keyfile must not be distributed in any way. A TeX DVI file and a postscript file of the manual are also supplied with the registered version.

The fee for a Filer registration is

- 40.- DM (D-Mark),
- 40.- SFr (Schweizer Franken),
- 180.- FF (French Francs),
- 30.- US\$ (US Dollar)

or an equivalent amount of US\$ 35.- (Thirty-five!) in any other currency. "Thirty-five" because I have to visit my bank and sell your foreign currency which is kind of expensive here in Germany. If you think your bank does it cheaper, feel free to get some US\$ 30.- at home and send them.

The fastest, cheapest and easiest way to register is put the money together with the filled registration form into a letter and send it to

Matthias Scheler  
Schützenstraße 18  
33178 Borcheln  
FR Germany

Euro cheques (in DM) are also welcome, but please do not send any kind of foreign cheques since the bank charges outrageous fees from both the sender and the receiver. If you really feel you must use a such cheques, please include extra US \$10.-.

You can also transfer the money directly to my bank account, but beware: international transfers via banks are very expensive.

Stadtsparkasse Paderborn, BLZ 472 501 01, Konto 16168007

Attention: Paying via bank transfer does NOT mean that you get your registered version of Filer before paying.

---

In any case, be sure to provide me with your name, address, phone number and e-mail address for filing purposes. You can send this information via electronic mail, if you wish. This data will be stored and processed in electronic form. There is a sample registration form in the file "OrderForm".

All registrations will be handled as fast as possible and should be finished in about two weeks.

## 1.4 Filer.guide/IN\_PURPOSE

Introduction

\*\*\*\*\*

Purpose

=====

Filer is an Intuition controlled file manager for OS 2.04 and newer.

You can use Filer for copying, deleting and renaming files or creating directories. You can create, list and extract archives with it or use it as an universal viewing tool. Filer even is able to control your whole system as a replacement for the Workbench.

Some of Filer's features are:

- \* font-sensitive, resizable and Style Guide compliant GUI
- \* ARexx interface
- \* highly configurable
- \* keyboard support
- \* class support
- \* archive support

## 1.5 Filer.guide/IN\_SYSR

System requirements

=====

Filer requires Amiga operating system version 2.04 or better. To be able to use all features version 3.1 is recommended.

Kickstart 1.3 is not supported; this operating system is considered obsolete.

Filer requires at least one megabyte of RAM to run because Workbench

2.04 needs a lot of memory for itself. A hardisk or a faster CPU is not required but increase performances and comfort, of course.

## 1.6 Filer.guide/IN\_INST

Installation  
=====

It is really easy to install Filer:  
Just copy the binary and the icon (called "Filer" and "Filer.info") to any directory and that's it.

Also, it's not a bad idea to copy the supplied configuration file (called "Examples/Filer.RC") to the "S:" directory of your system drive.

## 1.7 Filer.guide/UF\_MW

Usage of Filer  
\*\*\*\*\*

The Main window  
=====

The main window of Filer contains the following control elements:

Status Bar	The long gadget at the top
Directory Listview	The two large listviews
Device Listview	The small listview in the middle
Path Gadget	The two large string gadgets
Copy Gadget	The two arrows between the string gadgets
CD Gadget	The two gadgets labeled "CD"
Parent Gadget	The two gadgets labeled "/"
Cache Gadget	The two gadgets labeled "?"
Pattern Gadget	The two gadgets labeled "Pattern"
All Gadget	The two gadgets labeled "All"
None Gadget	The two gadgets labeled "None"
Command Panel	The 4 rows of gadgets at the bottom

## 1.8 Filer.guide/UF\_MW\_SBAR

## Status Bar

-----

The Status Bar is used to display information about operations performed by Filer. E.g. if you copy files the one which is currently copied will be shown there.

Events shown in the Status Bar are saved in a history buffer. You can look at this history buffer after clicking on the Status Bar. The viewer used for this can be configured with the VIEWER command in the config file.

The contents of the Status bar and the history buffer may be modified by the ARexx commands STATUS, HISTORY and CLEARHISTORY.

## 1.9 Filer.guide/UF\_MW\_LV

### Directory Listview

-----

The directory listviews are used to display the contents of the directories presently loaded. Each listview has two scrollers with additional arrows, one at the right side for vertical and one at the bottom for horizontal movement. Above each directory listview is a small status line. If no files or directories are selected you can see the free disk space otherwise the number and the total number of bytes occupied by the selected files and directories is shown. The directory listview with the highlighted status line is the active one. You can activate the other one by clicking on its status line. The active one is always the one which is affected by operations like deleting or renaming files. For copy or move operations it is the source directory.

You can select files and directories by clicking on them. Drag select is supported, too. Doubleclicking on a directory loads this directory, doubleclicking on a file starts a Auto Action for this file.

The format used for displaying data in this listview can be controlled with the FORMAT command in the config file.

You can activate the inactive listview with the ARexx command OTHERDIR.

## 1.10 Filer.guide/UF\_MW\_DLV

### Device Listview

-----

The third listview in the middle of the main window is the device listview. Real devices (e.g. "DF0:", "DH0:" or "RAM:") are displayed in the upper half, assigns and volumes in the lower half.

---

Filer has a filter for the device listview which is controlled by the HIDE command in the config file.

By double clicking on one of the devices shown in the listview you can load it into the active directory listview. If you want to do this with a single click turn the DOUBLECLICK option off.

## 1.11 Filer.guide/UF\_MW\_PATH

Path Gadget

-----

The two path gadgets below the directory listviews contain the pathnames of the directories actually loaded. You can load a directory by entering its pathname into one of these gadgets.

The path gadget of the active directory listview can be activated by the ARexx command ENTERDIR.

## 1.12 Filer.guide/UF\_MW\_COPY

Copy Gadget

-----

These gadgets are used to copy the contents of one directory listview to the other. The left arrow copies from right to left, the right one from left to right. Only the contents of the listviews is modified, the directories themselves are not affected.

## 1.13 Filer.guide/UF\_MW\_CD

CD Gadget

-----

With the CD gadget you can change the current directory of the directory listviews. After you activated this gadget by clicking on it you can select any directory or device shown in ANY of the three listviews. The selected directory will be loaded in the listview above the CD gadget. To cancel this function click on the gadget again.

## 1.14 Filer.guide/UF\_MW\_PAR

### Parent Gadget

-----

With the parent gadget you can load the parent directory of the current one into the directory listview above this gadget. If there is no parent directory nothing happens.

The same can be done with the ARexx command PARENTDIR.

## 1.15 Filer.guide/UF\_MW\_CACHE

### Cache Gadget

-----

With the cache gadget you can open the cache selection requester. This requester shows all directories which are currently in Filer's cache. You can select one of these and it will be copied into the directory listview above the cache gadget.

The order of the directories in this requester can be controlled with the SORTCACHE flag in the config file.

The same can be done with the ARexx command SELECTCACHE.

## 1.16 Filer.guide/UF\_MW\_PAT

### Pattern Gadget

-----

With the pattern gadget you can open the pattern matching requester. Any AmigaDOS pattern can be entered and all matching files and directories in the directory listview above the pattern gadget will be selected.

The same can be done with the ARexx command PATTERN.

## 1.17 Filer.guide/UF\_MW\_ALL

### All Gadget

-----

Selecting one of these gadgets will select all entries in the directory listview above.

The same can be done with the ARexx command ALL.

---



## 1.18 Filer.guide/UF\_MW\_NONE

None Gadget

-----

Selecting one of these gadgets will deselect all entries in the directory listview above.

The same can be done with the ARexx command NONE.

## 1.19 Filer.guide/UF\_MW\_PANEL

Command Panel

-----

At the bottom of Filer's main window is the command panel, four rows of eight gadgets each. With these gadgets you can start any kind of operation e.g. copying or deleting files. Filer supports upto to eight configurations for the panel at the same time, but you can NOT switch between them with the unregistered version.

The contents of the command panel can be changed with the BUTTON or XBUTTON command in the config file.

One of the eight panel configurations can be activated with the ARexx command BANK.

## 1.20 Filer.guide/UF\_MENU

Pull-Down Menus of the Main window

=====

Filer's main window has these menus:

- \* Project

- \* About...

- Open the copyright requester

- \* Clean Up Memory

- Clean the directory cache and flush all unused libraries and devices.

- \* Quit

- If the STAY option is on goto sleep mode else quit Filer.

- \* Quit & Die

Quit Filer even if STAY option is on.

- \* Preferences

- \* Miscellaneous

- \* Stay Resident

- Set the STAY option.

- \* Silent

- Set the NOASK option.

- \* Extend Path

- Set the EXTEND option.

- \* Cache

- Set the CACHE option.

- \* Doubleclick

- Set the DOUBLECLICK option.

- \* Deselect

- Set the DESELECT option.

- \* Use ExAll()

- Set the EXALL option.

- \* Automatic Requesters

- Set the AUTOREQUESTER option.

- \* Memory Display

- Set the MEMWINDOW option.

- \* Weekdays

- Set the WEEKDAYS option.

- \* reversed Colors

- Set the REVERSECOLORS option.

- \* Sort Cache

- Set the SORTCACHE option.

- \* Select Icon

---

Set the SELECTICON option.

- \* Click on Border for Parentdir

Set the BORDERPARENTDIR option.

- \* Update Listview

Set the LISTVIEWUPDATE option.

- \* Read directory after Diskchange

Set the DISKCHANGE option.

- \* Copy

- \* Flags

Set the COPYFLAGS option.

- \* Date

Set the COPYDATE option.

- \* Comment

Set the COPYNOTE option.

- \* Overwrite

Set the OVERWRITE option.

- \* Check free Diskspace

Set the CHECKFREE option.

- \* Format

- \* Verify

Set the VERIFY option.

- \* FFS

Set the FFS option.

- \* International

Set the INTERNATIONAL option.

- \* Directory Cache

Set the DIRCACHE option.

- \* Sound Playback

- \* Loop
-

Set the LOOP option.

\* Filter

Set the FILTER option.

\* Display Mode...

Open the Display Mode requester.

\* Snapshots the position and size of the main window on a public screen.

\* Font...

Open the Font requester

\* Screen Font...

Open the Screen Font requester

\* Colors...

Open the Color requester.

\* Viewer

\* Display Mode...

Set the Display Mode for the internal text viewer.

\* Font...

Set the Font for the internal text viewer.

\* External...

Switch to an external text viewer.

\* Command Buttons...

\* Contents...

Set the contents of the buttons in the command panel.

\* Font...

Set the font for the command panel.

\* Display Format...

Set the Display Format for the directory listviews.

\* Editor...

Set the Editor.

---

- \* Hide...

Set the filter for the device listview.

- \* Edit

Edit the Preferences with the external text editor.

- \* Load

Load the Preferences.

- \* Save

Save the Preferences.

## 1.21 Filer.guide/UF\_REQ

### Filer's Standard Requesters

=====

Filer has four standard requesters for various purposes:

- \* The message requester

The message requester is used to display any kind of messages, especially error messages. An example for this requester is the copyright requester. You can close this requester by clicking on the gadget or pressing a key. If the AUTOREQUESTER option is on deactivating the window will be enough, too.

A message requester can be opened with the ARexx command ALERTBOX.

- \* The decision requester

The decision requester is used if Filer has a question to the user. This requester has two gadgets labeled "Yes" and "No". For a positive decision click on "Yes" or press Y, y, RETURN or SPACE, for a negative one click on "No" or press N, n or ESC. If the AUTOREQUESTER option is on, deactivating the window will be considered as a negative answer, too.

A decision requester can be opened with the ARexx command QUESTBOX.

- \* The string requester

The string requester is used if Filer requires an textual input. You can close this requester by finishing your input with RETURN or clicking on one of the two or three gadgets. The rightmost gadget is always used to abort a function.

A string requester can be opened with the ARexx command GETSTRING.

---

\* The operation requester

The operation requester appears during many operations e.g. copying or deleting files. It is used to show the progress of the current operation and to abort it by pressing ESC or clicking on the "Abort" gadget.

## 1.22 Filer.guide/UF\_FN

Filer's builtin functions

=====

Filer has these internal functions:

Copy	Copy files and directories.
Copy As	Copy files and directories with new names.
Delete	Delete files and directories.
Disk Usage	Check free disk space.
Duplicate	Duplicate files and directories.
FileNote	Modify filenotes.
MakeDir	Create directories.
MakeLink	Create links.
Move	Move files and directories.
Move As	Move files and directories with new names.
Protect	Modify protection flags.
Relabel	Relabel volumes.
Rename	Rename files and directories.
Change Owner	Change owner of files.
Change Group	Change group of files.
Action	Automatically handle files.
Hex Dump	View hexdumps of files.
Run	Execute programs and shell scripts.
Type	View text files.
Add	Add files to archives.
Add & Delete	Move files to archives.
Extract	Extract files from archives.
List	List contents of archives.
Edit	Edit text files.
Format	Format disk or harddisks.
Search	Search for files or directories.
Quit	Quit Filer.

## 1.23 Filer.guide/UF\_FN\_COPY

### "Copy" function

-----

This function copies all selected files and directories in the active directory listview to the directory shown in the other listview, Directories are recursively scanned and recreated on the target side.

Dates, protection flags and filenotes may be copied, too, depending on the configuration.

You may abort this function via the operation requester.

This function can be invoked with the ARexx command COPY.

## 1.24 Filer.guide/UF\_FN\_COPYAS

### "Copy As" function

-----

This function copies all selected files and directories in the active directory listview to the directory shown in the other listview. For each selected entry you will be prompted for the name of the copy. Directories are recursively scanned and recreated on the target side.

Dates, protection flags and filenotes may be copied, too, depending on the configuration.

You may abort this function via the operation requester.

This function can be invoked with the ARexx command COPY AS.

## 1.25 Filer.guide/UF\_FN\_DELETE

### "Delete" function

-----

This function deletes all selected files and directories in the active directory listview. If the NOASK option is turned on, a warning requester will appear before the operation is started.

If Filer hits a delete protected file, a requester will appear. You have three options: Delete this file anyway, delete all protected files or abort the operation.

You may abort this function via the operation requester.

This function can be invoked with the ARexx command DELETE.

---

## 1.26 Filer.guide/UF\_FN\_DU

"Disk Usage" function  
-----

This function calculates the disk usage of everything selected in the active directory listview. These directories will not become deselected.

You may abort this function via the operation requester.

This function can be invoked with the ARexx command DU.

## 1.27 Filer.guide/UF\_FN\_DUPLICATE

"Duplicate" function  
-----

This function duplicates all selected files and directories in the active directory listview. For each file or directory you will be asked for a new name via a requester and Filer will create a copy with the new name in the same directory.

Dates, protection flags and filenotes may be copied, too, depending on the configuration.

You may abort this function via the operation requester.

This function can be invoked with the ARexx command DUPLICATE.

## 1.28 Filer.guide/UF\_FN\_FILENOTE

"FileNote" function  
-----

This function modifies the filenotes of all selected files and directories in the active FILENOTE. You can change every filenote with a string requester. You can leave out files by selecting the "Skip" gadget.

You may abort this function by selecting the "Cancel" gadget.

This function can be invoked with the ARexx command FILENOTE.

## 1.29 Filer.guide/UF\_FN\_MAKEDIR



"MakeDir" function

-----

This function opens a requester to ask for the name of a new directory. After the directory is successfully created, you will be asked whether an icon should be created for it. Of course, Filer will use the system default drawer icon.

You may abort this function by selecting the "Cancel" gadget.

This function can be invoked with the ARexx command MAKEDIR.

### 1.30 Filer.guide/UF\_FN\_MAKELINK

"MakeLink" function

-----

This function opens a requester to ask for the name of the new link. If you acknowledge the input by pressing RETURN or select the "Hard" gadget a hardlink will be created. For a softlink select the "Soft" gadget. After the name of the link was entered a file requester for selecting the link target will appear.

You may abort this function by selecting the "Cancel" gadget.

This function can be invoked with the ARexx command MAKELINK.

### 1.31 Filer.guide/UF\_FN\_MOVE

"Move" function

-----

This function moves all selected files and directories in the active directory listview to the directory shown in the other listview, existing files are overwritten. Directories are recursively scanned and recreated on the target side. Moving a file or directory means to copy it and delete the original file.

Dates, protection flags and filenotes may be copied, too, depending on the configuration.

You may abort this function via the operation requester.

This function can be invoked with the ARexx command MOVE.

### 1.32 Filer.guide/UF\_FN\_MOVEAS

"Move As" function

-----

This function moves all selected files and directories in the active directory listview to the directory shown in the other listview. For each selected entry you will be prompted for the name in the target directory. Directories are recursively scanned and recreated on the target side. Moving a file or directory means to copy it and delete the original file.

Dates, protection flags and filenotes may be copied, too, depending on the configuration.

You may abort this function via the operation requester.

This function can be invoked with the ARexx command MOVE AS.

### 1.33 Filer.guide/UF\_FN\_PROTECT

"Protect" function

-----

This function opens a requester for setting the protection flags. If a flag is toggled to Set it will be set, Unset means to clear it and Ignore will left in unchanged. After clicking the "Okay" gadget the protection flags of all selected files and directories in the active directory listview will be changed.

You may abort this function by selecting the "Cancel" gadget.

This function can be invoked with the ARexx command PROTECT.

### 1.34 Filer.guide/UF\_FN\_RELABEL

"Relabel" function

-----

This function opens a requester to ask for a new volume name. After you entered it, the volume will be relabeled.

You may abort this function by selecting the "Cancel" gadget.

This function can be invoked with the ARexx command RELABEL.

### 1.35 Filer.guide/UF\_FN\_RENAME

### "Rename" function

-----

For each selected file and directory in the active directory listview a requester is opened. You can enter a new name into this requester and the file or directory will be renamed. You can leave out files by selecting the "Skip" gadget.

You may abort this function by selecting the "Cancel" gadget.

This function can be invoked with the ARexx command RENAME.

## 1.36 Filer.guide/UF\_FN\_CHOWN

### "Change Owner" function

-----

This function opens a requester to ask for the name of an Envoy user. You can enter a decimal number, too. Afterwards the owner of all selected files and directory will be changed to that one.

This function will not work with the version 37 or older of the ROM FileSystem. Newer versions of the ROM FileSystems will store the owner but ignore it otherwise. Access control is only active if you export the volume via Envoy.

You may abort this function via the operation requester.

## 1.37 Filer.guide/UF\_FN\_CHGRP

### "Change Group" function

-----

This function opens a requester to ask for the name of an Envoy group. You can enter a decimal number, too. Afterwards the group of all selected files and directory will be changed to that one.

This function will not work with the version 37 or older of the ROM FileSystem. Newer versions of the ROM FileSystems will store the group but ignore it otherwise. Access control is only active if you export the volume via Envoy.

You may abort this function via the operation requester.

## 1.38 Filer.guide/UF\_FN\_ACTION

### "Action" function

-----

This function performs an Auto Action on all selected files directories in the active directory listview.

You may abort this function via the operation requester.

This function can be invoked with the ARexx command ACTION.

## 1.39 Filer.guide/UF\_FN\_HEXDUMP

### "Hex Dump" function

-----

This function shows a hexdump of all selected files in the active directory listview using the internal text viewer.

You may abort this function by pressing SHIFT ESC in the text viewer window.

This function can be invoked with the ARexx command HEXDUMP.

## 1.40 Filer.guide/UF\_FN\_RUN

### "Run" function

-----

This function executes all selected files in the active directory listview. You can even execute shell scripts this way.

You may abort this function selecting the close gadget of the shell window.

This function can be invoked with the ARexx command RUN.

## 1.41 Filer.guide/UF\_FN\_TYPE

### "Type" function

-----

This function shows all selected files in the active directory listview as text files.

You may select the text viewer with the VIEWER command in the config file.

---

You may abort this function by pressing SHIFT ESC in the text viewer window.

This function can be invoked with the ARexx command TYPE.

## 1.42 Filer.guide/UF\_FN\_ADD

"Add" function

-----

This function is used to create archives or to add files to existing archives.

To create a new archive select all files and directories to add and no file in the target directory listview. You will have to enter the name for the new archive in a requester.

To add files to an existing archive select the archive in one directory listview and the files and directories in the other listview. This listview with the files must be the active one if you invoke this function.

Archive formats can be added with the ARCDEF command in the config file.

This function can be invoked with the ARexx command ADD.

## 1.43 Filer.guide/UF\_FN\_ADDDEL

"Add & Delete" function

-----

This function is used to create archives or to add files to existing archives. The files will be deleted after being archived.

To create a new archive select all files and directories to add and delete and no file in the target directory listview. You will have to enter the name for the new archive in a requester.

To add files to an existing archive and delete them select the archive in one directory listview and the files and directories in the other listview. This listview with the files must be the active one if you invoke this function.

After all files have been archived successfully they will be deleted. If the NOASK option is turned on, a warning requester will appear before that.

Archive formats can be added with the ARCDEF command in the config file.

---

This function can be invoked with the ARexx command ADD DELETE.

## 1.44 Filer.guide/UF\_FN\_EXTRACT

"Extract" function

-----

This function extracts files from all selected archives in the active directory listview. If a file is no known archive format the function will abort with a requester. You can enter a pattern for the files to be extracted.

Archive formats can be added with the ARCDEF command in the config file.

This function can be invoked with the ARexx command EXTRACT.

## 1.45 Filer.guide/UF\_FN\_LIST

"List" function

-----

This function lists the contents of all selected archives in the active directory listview. If a file is not of a known archive format the function will abort with a requester.

Archive formats can be added with the ARCDEF command in the config file.

This function can be invoked with the ARexx command LIST.

## 1.46 Filer.guide/UF\_FN\_EDIT

"Edit" function

-----

This function edits all selected files with an external text editor. After the last file has been edited the directory is checked for changes and reread if necessary.

You may select the text editor with the EDITOR command in the config file.

This function can be invoked with the ARexx command EDIT.

---

## 1.47 Filer.guide/UF\_FN\_FORMAT

"Format" function  
-----

Attention:

This function is extremely dangerous.  
You may destroy vital data.  
Be careful!

This function opens the format requester. At the left side you can see a listview with all available trackdisk like devices. Select one by clicking on it or use the cursor keys to move the cursor and SPACE or RETURN to select.

At the top of the window you see a large box which is used to show information about the select device. You can see the drive's name and size, the FileSystem used, the exec device and unit number there.

Below the information box you find four setup gadgets. The "Verify" gadget is used to turn verify mode on and off for low level formats. "FastFileSystem", "International Mode" and "Directory Cache" are used to specify the filesystem type used for the new volume. These settings are not available for custom filesystems like CrossDOS.

At the bottom of the format requester you can see the four operation gadgets. "Format" is used to format a volume, "Quick Format" just creates a new directory structure. "Install" writes a standard bootblock to disk (only supported for standard filesystem on disk-like devices). "Cancel" is used to close the requester and return to the main window.

The default settings for the "Verify", "FastFileSystem", "International Mode" and "Directory Cache" gadgets can be modified with the SET command in the config file.

You can close the requester by selecting "Cancel" or pressing ESC.

This function can be invoked with the ARexx command FORMAT.

## 1.48 Filer.guide/UF\_FN\_SEARCH

"Search" function  
-----

This function asks for a search pattern first. You can enter any AmigaDOS pattern now and Filer will search all selected directories in the active directory listview for Files which names match this pattern. During the scan a normal CON: window is used to display the results. The parameters of this window can be changed with the CONHANDLER in the configuration file.

You may abort this function by pressing ESC in the search window.

---

This function can be invoked with the ARexx command SEARCH.

## 1.49 Filer.guide/UF\_FN\_QUIT

"Quit" function

-----

This function will quit Filer. If the NOASK option is turned off a requester appears to verify you're not kidding. If the STAY option is turned on Filer will free most of its resources and go to sleep mode waiting for reactivation via ARexx e.g. from the Filer Commodity.

This function can be invoked with the ARexx command QUIT.

## 1.50 Filer.guide/UF\_KBD

Keyboard control

=====

Important:

Since version 3.10 Filer is controllable completely through keyboard use. Filer has no builtin default key map. All key shortcuts must be defined in the configuration file. Thus, the keys described in this manual work only if you use the definitions from the supplied example configuration.

Key control is available for these topics:

- Basic control
- Cursor support
- Listview movement
- Command shortcuts
- Directory shortcuts

## 1.51 Filer.guide/UF\_KBD\_BASIC

Basic control

-----

TAB

activate the other directory listview.

RETURN

activate the path gadget of the active directory listview.

+



select all files and directories in the active directory listview.

-

deselect all files and directories in the active directory listview.

HELP

open the pattern matching requester.

/

load the parent directory of directory in the active directory listview.

?

open the cache selection requester.

## 1.52 Filer.guide/UF\_KBD\_CURSOR

Cursor support

-----

You can move the cursor in the active directory listview with these keys:

SPACE

(de)select entry under the cursor and move cursor one line down

ENTER

perform auto action for entry under cursor

CURSOR-UP

cursor one line up

CURSOR-DOWN

cursor one line down

SHIFT CURSOR-UP

cursor one page up

SHIFT CURSOR-DOWN

cursor one page down

CTRL CURSOR-UP

cursor to top

CTRL CURSOR-DOWN

cursor to bottom

## 1.53 Filer.guide/UF\_KBD\_LV

---

### Listview movement

-----

You can scroll the active directory listview with these keys:

- 8       one line up.
- 2       one line down.
- 9       one page up.
- 3       one page up.
- 7       to the top.
- 1       to the bottom.
- 4       one character left.
- 6       one character right.

## 1.54 Filer.guide/UF\_KBD\_CMD

### Command shortcuts

-----

- a       Invoke "Add" function.
  - c       Invoke "Copy" function.
  - c       Invoke "Delete" function.
  - e       Invoke "Edit" function.
  - f       Invoke "Format" function.
  - h       Invoke "Hex Dump" function.
  - k
-

	Invoke "MakeDir" function.
l	Invoke "Relabel" function.
m	Invoke "Move" function.
n	Invoke "FileNote" function.
o	Invoke "Action" function.
p	Invoke "Protect" function.
q	Invoke "Quit" function.
R	Invoke "Rename" function.
s	Invoke "Search" function.
T	Invoke "Type" function.
u	Invoke "DU" function.
x	Invoke "Extract" function.

## 1.55 Filer.guide/UF\_KBD\_DIR

Directory shortcuts

-----

F1	Read directory "SYS:" to active directory listview.
SHIFT F2	Read directory "SYS:" to inactive directory listview.
F1	Read directory "T:" to active directory listview.
SHIFT F2	Read directory "T:" to inactive directory listview.

---

## 1.56 Filer.guide/UF\_ACT

Auto Action  
=====

Filer supports automatic handling of Files, so called "Auto Actions". An auto action for a single file can be started by doubleclicking on that file. To do this for all selected files use the ACTION function.

During an Auto Action Filer checks files in this order:

- \* Known archive format: List archive's contents.

Archive formats can be added with the ARCDEF command.

- \* Known class: Start the specified external tool.

Classes can be added with the CLASS or REXXCLASS command.

- \* IFF file: use internal loader if type is known otherwise abort.

- \* AmigaOS executable: Execute it.

Shell scripts will not be executed but viewed. To execute a script use the RUN function.

- \* Textfile: View it.

The text viewer can be set with the VIEWER command.

The Auto Action is finished after the first successful check. If all these checks fail an error requester appears.

## 1.57 Filer.guide/UF\_TEXT

Filer's internal Text viewer  
=====

Filer has an internal text viewer for viewing text files and hexdumps. The text viewer features high display speed, ANSI support and searching (not in hexdump mode). The text viewer does no hardware banging and therefore works fine with the Workbench emulations supplied with some graphic cards.

The text viewer can be controled with these keys:

CURSOR-UP / 8  
one line up.

CURSOR-DOWN / 2  
one line down.

---

SHIFT CURSOR-UP / 9  
one page up.

SHIFT CURSOR-DOWN / 3  
one page down.

CTRL CURSOR-UP / 7  
to the top

CTRL CURSOR-DOWN / 1  
to the bottom

F / S  
opens the search requester.

N  
repeats the search.

ESC / 5  
exits the textviewer.

SHIFT ESC  
exits textviewer and aborts if Filer's TYPE function was used.

## 1.58 Filer.guide/UF\_CF

Filer's Configuration File  
=====

Filer uses an text file for it's configuration. I prefer this method because you can modify such a file with your favourite editor. IFF files are nice for the programmer but bad for the user.

The configuration file is called "Filer.RC" and must be located in the "S:"-Directory - "ENV:" is already overloaded in my opinion. If you want to keep the configuration file in another directory use the environment variable "FILERCONFIG" to tell Filer the full filename. The configuration file will be read if Filer is started or the Load menu item is selected.

The configuration file contains commands, one per line. Comments are introduced with a "#" and end at the next newline. Commands and keywords are case insensitive. Parameters are usually separated by commas, string parameters are surrounded by quotes.

A piece of a configuration file may look like this:

```
#
# Sample Configuration for Filer 3.23
#

# display mode for Filer's screen

DISPLAY ""
```

```
SCREEN 2,DEFAULT,DEFAULT
WINDOW "",0,16,640,200
```

If you use the "Save" menu item filer will create a new configuration file from scratch. As a result of this user created comments will be overwritten.

Filer supports the following configuration commands:

User interface configuration:

```
DISPLAY
SCREEN
WINDOW
COLOR
RGB

FONT
PANELFONT
SCREENFONT
```

Program control configuration:

```
BUTTON
XBUTTON
KEY
CLOSEGADGET
STARTUP
SHUTDOWN
```

Global configuration:

```
SET
```

Miscellaneous flags:

```
COPYBUFFER
FORMAT
HIDE
HISTORY
```

Textviewer:

```
VIEWER
VIEWERDISPLAY
VIEWERFONT
```

Starting external programs:

```
ADDPATH
STACK
CONHANDLER
```

Integration of external programs:

```
EDITOR
```

---

```
ARCDEF
DEFARCEXT

IFFDEF
TOOL
REXXTOOL
CLASS
REXXCLASS
```

## 1.59 Filer.guide/UF\_CF\_DISPLAY

```
DISPLAY
-----
```

### SYNOPSIS

```
    DISPLAY "mode"
```

### DESCRIPTION

DISPLAY specifies the mode for Filer's screen. This can be either an empty string to clone the workbench, a screen mode name from the display database like "NTSC:HighRes Interlace" or "PubScreen" to open Filer on a public screen.

### ARGUMENTS

mode: a display mode keyword.

### DEFAULT

```
    DISPLAY ""
```

### SEE ALSO

SCREEN, WINDOW

## 1.60 Filer.guide/UF\_CF\_SCREEN

```
SCREEN
-----
```

### SYNOPSIS

```
    SCREEN depth,width,height
```

### DESCRIPTION

SCREEN is used to set the depth and dimension of the screen used by Filer. Filer currently supports screens with a minimum of 4 colors upto a maximum of 16 under OS 2.x and 256 under OS 3.x. The width and height parameter may be used to create screens with dimensions larger than the visible area. This is useful for autoscroll or Workbench emulation's without screen mode support. You can use DEFAULT for either width or height to get the default dimension from the display database.

## ARGUMENTS

depth : depth of the screen [2-8].  
width : width of the screen  
height: height of the screen

## DEFAULT

SCREEN 2, DEFAULT, DEFAULT

## SEE ALSO

DISPLAY, WINDOW

## 1.61 Filer.guide/UF\_CF\_WINDOW

## WINDOW

-----

## SYNOPSIS

WINDOW "name", left, top, width, height

## DESCRIPTION

WINDOW is used to set the public screen for Filer's window and the dimensions of this window. The first argument is the name of the public screen. If the string is empty the default public screen is used. The left and top arguments are used to specify the upper left corner of the window. You may use DEFAULT for one or both arguments as a synonym for the left or upper edge of the visible area of the screen. The width and height arguments are used to specify the size of the window. Again you may use DEFAULT as a synonym this time for the width or height of the visible area of the screen.

## ARGUMENTS

name : name of the public screen.  
left : left edge of the window.  
top : top edge of the window.  
width : width of the window.  
height: height of the window.

## DEFAULT

WINDOW "", DEFAULT, DEFAULT, DEFAULT, DEFAULT

## SEE ALSO

DISPLAY, SCREEN

## 1.62 Filer.guide/UF\_CF\_COLOR

## COLOR

-----

---



## SYNOPSIS

COLOR pen,RGB

## DESCRIPTION

COLOR is an obsolete command to set the colors for Filer's screen. It's only supported for compatibility reasons, use RGB instead.

## ARGUMENTS

pen: a pen number [0-15].  
RGB: a hexadecimal RGB value.

## SEE ALSO

RGB

## 1.63 Filer.guide/UF\_CF\_RGB

## RGB

---

## SYNOPSIS

RGB pen,red,green,blue

## DESCRIPTION

RGB sets the color for one of the pens of Filer's GUI. Pens 0-3 are always be used, the other 12 pens need to be allocated by a RGB command. The supplied RGB values select one color out of 16777216 possible colors. The best way to adjust these values is to use the internal color requester.

## ARGUMENTS

pen : pen number [0-15].  
red : hexadecimal RGB red value [0-FF].  
green: hexadecimal RGB green value [0-FF].  
blue : hexadecimal RGB blue value [0-FF].

## DEFAULT

RGB 0,A0,A0,A0  
RGB 1,00,00,00  
RGB 2,FF,FF,FF  
RGB 3,60,80,B0

## 1.64 Filer.guide/UF\_CF\_FONT

## FONT

----

## SYNOPSIS

FONT "name",size

## DESCRIPTION

FONT sets the font used for the Filer's listviews and the status bar. This font must NOT be a proportional font.

**ARGUMENTS**

name: name of the font.  
size: size of the font.

**DEFAULT**

FONT "topaz",8

**SEE ALSO**

PANELFONT, SCREENFONT

## 1.65 Filer.guide/UF\_CF\_PANELFONT

**PANELFONT**

-----

**SYNOPSIS**

PANELFONT "name",size

**DESCRIPTION**

PANELFONT sets the font used for the Filer's main panel (the command buttons in the main window). This font may be a proportional font.

**ARGUMENTS**

name: name of the font.  
size: size of the font.

**DEFAULT**

PANELFONT "topaz",8

**SEE ALSO**

FONT, SCREENFONT

## 1.66 Filer.guide/UF\_CF\_SCREENFONT

**SCREENFONT**

-----

**SYNOPSIS**

SCREENFONT "name",size

**DESCRIPTION**

SCREENFONT sets the font used as default font for Filer's screen. This font is used e.g. for window and screen titles or for buttons in all preference requesters. This font may be a proportional font.

**ARGUMENTS**

```
name: name of the font.  
size: size of the font.
```

## DEFAULT

```
SCREENFONT "topaz",8
```

## SEE ALSO

```
FONT, PANELFONT
```

## 1.67 Filer.guide/UF\_CF\_BUTTON

## BUTTON

-----

## SYNOPSIS

```
BUTTON row,column,"label","command"
```

## DESCRIPTION

BUTTON defines the look and purpose of one of the command buttons in the main panel. Row and column specify the position of this button. Label is the label printed on the button. It may contain underscores to mark a hotkey for the button (You must of course define this hotkey via a KEY command). Command is the command that is sent to Filer's ARexx interface if you click on this button.

## ARGUMENTS

```
row      : row of the button [0-3]  
column   : column of the button [0-7]  
label    : label of the button  
command: ARexx command for the button
```

## EXAMPLE

```
BUTTON 0,0,"_Copy","COPY"  
BUTTON 0,1,"_Delete","DELETE"
```

## SEE ALSO

```
XBUTTON, Filer's ARexx Interface
```

## 1.68 Filer.guide/UF\_CF\_XBUTTON

## XBUTTON

-----

## SYNOPSIS

```
XBUTTON bank,row,column,pen,"label","command"
```

## DESCRIPTION

XBUTTON is an eXtended version of BUTTON. It's used to specify buttons with different pens and for other banks. Filer supports 8 separate banks of command buttons for the main panel (see BANK).

The bank for the button belongs is specified by the bank argument. Pen is the number of the pen used for this button (pen colors are specified with RGB). The arguments of row, column, label and command work the same way as described in the section about the BUTTON command.

#### ARGUMENTS

```
bank      : number of the panel bank [0-7]
row       : row of the button [0-3]
column    : column of the button [0-7]
pen       : pen number for the label [0-15]
label     : label of the button
command:  ARexx command for the button
```

#### EXAMPLE

```
XBUTTON 0,0,0,1,"_Copy","COPY"
XBUTTON 0,0,1,2,"_Delete","DELETE"
```

#### SEE ALSO

BUTTON, Filer's ARexx Interface

## 1.69 Filer.guide/UF\_CF\_KEY

#### KEY

---

#### SYNOPSIS

```
KEY "description", "command"
```

#### DESCRIPTION

KEY defines a command key for Filer. If the specified key is pressed the command is sent to Filer's ARexx port. Although the commodities syntax is used for key descriptions these command keys are NO hotkeys, they are only active if Filer's window is activated.

#### ARGUMENTS

```
description: a key in commodity syntax description.
command    : ARexx command for the key.
```

#### EXAMPLE

```
KEY "c", "COPY"
KEY "d", "DELETE"
```

#### SEE ALSO

Filer's ARexx Interface

## 1.70 Filer.guide/UF\_CF\_CLOSEGADGET

## CLOSEGADGET

-----

### SYNOPSIS

CLOSEGADGET "command"

### DESCRIPTION

CLOSEGADGET defines the ARexx command which is sent to Filer's ARexx port if you click on the close gadget (available only in pubscreen mode) of the main window. You may use every command for this, but it should be either ICONIFY or QUIT.

### ARGUMENTS

command: ARexx command for the close gadget.

### DEFAULT

CLOSEGADGET "QUIT"

### SEE ALSO

QUIT, ICONIFY

## 1.71 Filer.guide/UF\_CF\_STARTUP

## STARTUP

-----

### SYNOPSIS

STARTUP "command"

### DESCRIPTION

STARTUP defines the ARexx command executed at startup time. It will not be executed if Filer is uniconified.

### ARGUMENTS

command: startup ARexx command.

### DEFAULT

STARTUP ""

### SEE ALSO

SHUTDOWN

## 1.72 Filer.guide/UF\_CF\_SHUTDOWN

## SHUTDOWN

-----

### SYNOPSIS

SHUTDOWN "command"

---

## DESCRIPTION

SHUTDOWN defines the ARexx command executed at shutdown time. It will not be executed if Filer is iconified.

## ARGUMENTS

command: shutdown ARexx command.

## DEFAULT

SHUTDOWN ""

## SEE ALSO

STARTUP

## 1.73 Filer.guide/UF\_CF\_SET

## SET

---

## SYNOPSIS

SET "variable",ON|OFF

## DESCRIPTION

SET is used to modify miscellaneous parameters for Filer. Variable is the name of the parameter and ON or OFF the value for it. The following variables are supported:

\* EXTEND

This flag controls the path extension. If the path extension is active all pathname in the active directory listview will be "extended". E.g. if you change to the directory "L:" you'll get something like "System3.0:L" (depends on the name of your system partition), "DF0:" would be expanded to the volume name of the disk in drive 0. There are some bogus filesystems which cause problems with this option e.g. older versions of SerNet or the freely distributable version of MessyDOS (MSH). So if you use one of them turn the option off.

\* CACHE

This flag controls the directory cache. Filer has a own directory cache for faster operation especially on floppy disk. If you are low on memory you can switch the cache off.

\* DOUBLECLICK

This flag controls the device listview in the main window. If it is turned on you need to double click on an entry in the device listview to load it into the active directory listview. If it is turned off a single click suffices. This flag had a different purpose in older versions of Filer.

\* NOASK

---

This flag turns off some user requests, e.g. the warning if you invoke DELETE. Beginners should leave this option turned off.

★ STAY

If this flag is turned on Filer will stay resident in memory after you use QUIT. This way in can be reactivated quickly by FilerCom or via ARexx.

★ DESELECT

If Filer stores a directory in its directory cache all entries are marked as unselected. If you want your selections to stay active turn this option off.

★ EXALL

With this option you control how Filer reads directories. If it is turned on Filer uses the fast ExAll() call. This call has lots of bugs under OS <3.1, so you'll better turn it off under OS 2.x.

★ AUTOREQUESTER

To allow fast usage Filer's standard requesters can be configured to vanish if you deactivate their window. This causes trouble with some weird commodities e.g. any kind of auto window activators. If you are not using one of the commodities it's good idea to turn this option on.

★ MEMWINDOW

This flag controls Filer's memory usage window. If it is turned on the window appears. If Filer's running in pubscreen mode there's no window, the memory usage is displayed in the title bar of the main window.

★ WEEKDAYS

Normally Filer displays all dates of files or directories in standard AmigaDOS format "DD-MMM-YY". If you turn this option on Filer will also use synonyms (e.g. yesterday) and weekdays (monday, tuesday, etc.).

★ REVERSECOLORS

Older versions of Filer displayed directories in black and files in white. The actual version displays directories in white and files in black like the ASL filerequester does. If you want the old look switch this option on.

★ SORTCACHE

In the cache window the directories are displayed in a last-in-at-top order per default. If you turn this option on

they will be sorted alphabetically.

★ SELECTICON

If you switch this flag on Filer will select or deselect the icon file automatically in the directory listview if you select or deselect a single file or directory.

★ BORDERPARENTDIR

If you switch this flag on you will active two invisible parent gadgets one at the left border and one at the right border of the main window. Both are located next to the directory listviews.

★ LISTVIEWUPDATE

If this flag is turned on Filer will keep its directory listviews uptodate while perfoming operations. While deleting files vanish immediately, while copying they appear immediately on the target side. The required graphic operations will of course slow down the operation a little bit.

★ DISKCHANGE

This flag controls how Filer reacts on diskchanges. If you insert a new volume in a device which is shown in one directory listviews the root directory of this volume is automatically read if you have turned on this option.

★ COPYFLAGS

If this flag is turned on Filer copies protection flags of files or directories. This is also the default behaviour of the CLI command "Copy".

★ COPYDATE

If this flag is turned on Filer copies the creation date of files or directories. The CLI command "Copy" does the same if you use the "DATES" option.

★ COPYNOTE

If this flag is turned on Filer copies filenotes of files or directories. The CLI command "Copy" does the same if you use the "COM" option.

★ CHECKFREE

If this option is active Filer checks the target volume from the required free disk space before copying, moving or duplicating files or directories. If there is not enough disk space left the user is requested to confirm the operation.

★ OVERWRITE

---



If this option is inactive Filer ask before overwriting a file while copying, duplicating or moving files.

★ VERIFY

This flag is the preset for the "Verify" gadget in the format requester.

★ FFS

This flag is the preset for the "Fast File System" gadget in the format requester.

★ INTERNATIONAL

This flag is the preset for the "International" gadget in the format requester.

★ DIRCACHE

This flag is the preset for the "Directory Cache" gadget in the format requester. This flag is always turned off under OS <3.0.

★ LOOP

If you don't want that the 8SVX player repeats samples in a loop switch this flag off.

★ FILTER

If this flag is switched off the 8SVX player will disable the audio filter while playing samples.

#### ARGUMENTS

variable: name of the variable to set.  
boolean : either ON or OFF.

#### DEFAULT

SET EXTEND ON  
SET CACHE ON  
SET DOUBLECLICK ON  
SET COPYFLAGS ON  
SET COPYDATE ON  
SET COPYNOTE ON  
SET OVERWRITE OFF  
SET CHECKFREE OFF  
SET NOASK OFF  
SET STAY OFF  
SET DESELECT ON  
SET EXALL ON  
SET AUTOREQUESTER OFF  
SET MEMWINDOW ON  
SET WEEKDAYS OFF  
SET REVERSECOLORS OFF  
SET SORTCACHE OFF

```
SET SELECTICON OFF
SET BORDERPARENTDIR OFF
SET LISTVIEWUPDATE OFF
SET DISKCHANGE OFF
SET VERIFY ON
SET FFS OFF
SET INTERNATIONAL OFF
SET DIRCACHE OFF
SET LOOP ON
SET FILTER ON
```

## 1.74 Filer.guide/UF\_CF\_COPYBUFFER

COPYBUFFER

-----

### SYNOPSIS

COPYBUFFER bytes,ON|OFF

### DESCRIPTION

COPYBUFFER controls the size and memory type of the buffer used for copying files. Bytes is the maximum buffer size in bytes. You may increase this value to a very high value, because Filer will halve the buffer size automatically until enough free memory is available. Too large values may nevertheless cause problems with some filesystems, especially network filesystem. The second argument, a boolean value, is used to control the memory type of Filer's copy buffer. If it is turned on Filer allocates its buffer in so called "24 bit DMA memory". "24 bit DMA memory" is memory which can be used by a Zorro II card for DMA transfer. Using this type of memory makes copying of files on e.g. an A4000/040 with an A2091 hostadapter.

### ARGUMENTS

bytes : size of the copy buffer in bytes.  
24 bit DMA: either ON or OFF.

### DEFAULT

COPYBUFFER 131072,OFF

## 1.75 Filer.guide/UF\_CF\_FORMAT

FORMAT

-----

### SYNOPSIS

FORMAT "description"

### DESCRIPTION

FORMAT controls the format used for the directory listviews. Each

character represents a field to show in the display. The following characters are valid:

- \* "N": name
- \* "T": type (FI, DI, SL or HL)
- \* "L": length in bytes
- \* "P": protection flags
- \* "D": date
- \* "X": extended info (filenote or softlink target)
- \* "F": extended flags (Envoy)
- \* "O": owner (Envoy)
- \* "G": group (Envoy)

The format description is not case sensitive. The values for owner and group may be unavailable under OS 2.x.

#### ARGUMENTS

description: format description (see above).

#### DEFAULT

FORMAT "NLPTDX"

## 1.76 Filer.guide/UF\_CF\_HIDE

#### HIDE

----

#### SYNOPSIS

HIDE "device:"

#### DESCRIPTION

HIDE is used to declare AmigaDOS devices, volumes and assigns which will not show up in the device listview. This command may be used multiple times.

#### ARGUMENTS

device: a AmigaDOS device.

#### EXAMPLE

```
HIDE "CLIPS:"  
HIDE "ENV:"  
HIDE "ENVARC:"
```

## 1.77 Filer.guide/UF\_CF\_HISTORY

### HISTORY

-----

### SYNOPSIS

HISTORY bytes

### DESCRIPTION

HISTORY sets the size of Filer's history buffer. This buffer is used to log all actions done by Filer. It can be shown by clicking on the status bar.

### ARGUMENTS

bytes: history buffer size in bytes.

### DEFAULT

HISTORY 16384

### SEE ALSO

Status Bar

## 1.78 Filer.guide/UF\_CF\_VIEWER

### VIEWER

-----

### SYNOPSIS

VIEWER "command"

### DESCRIPTION

VIEWER is used to install an external textviewer for Filer. Command is a CLI command which will be called as textviewer. It should contain a "%s" to mark the location where the filename will be inserted. An empty command will activate the internal textviewer.

### ARGUMENTS

command: CLI command to call external textviewer.

### DEFAULT

VIEWER ""

### EXAMPLE

VIEWER "MultiView %f PUBSCREEN %p"

### SEE ALSO

Starting external programs

---

## 1.79 Filer.guide/UF\_CF\_VIEWERDISPLAY

VIEWERDISPLAY

-----

### SYNOPSIS

VIEWERDISPLAY "mode"

### DESCRIPTION

VIEWERDISPLAY specifies the mode for the screen of Filer's textviewer. This can be either an empty string to clone Filer's screen, a screen mode name from the display database like "NTSC:HighRes Interlace" or "PubScreen" to open the textviewer on Filer's screen.

### ARGUMENTS

mode: a display mode keyword.

### DEFAULT

VIEWERDISPLAY ""

## 1.80 Filer.guide/UF\_CF\_VIEWERFONT

VIEWERFONT

-----

### SYNOPSIS

VIEWERFONT "name",size

### DESCRIPTION

VIEWERFONT sets the font for Filer's internal textviewer. This font must NOT be a proportional font.

### ARGUMENTS

name: name of the font.  
size: size of the font.

### DEFAULT

VIEWERFONT "topaz",8

## 1.81 Filer.guide/UF\_CF\_ADDPATH

ADDPATH

-----

### SYNOPSIS

ADDPATH "directory"

### DESCRIPTION

ADDPATH is used to build Filer's CLI command search path. When

starting an external program e.g. a textviewer or an archiver  
Filer will search for the executable in all specified directories.  
This command may be used multiple times.

**ARGUMENTS**

directory: directory for the command search path.

**EXAMPLE**

```
ADDPATH "SYS:System"  
ADDPATH "SYS:Tools"  
ADDPATH "SYS:Utilities"
```

## 1.82 Filer.guide/UF\_CF\_STACK

**STACK**

-----

**SYNOPSIS**

STACK bytes

**DESCRIPTION**

STACK sets the stack size which Filer uses for calling external programs. Some programs need more than the standard 4000 bytes, so if you have some memory left 16384 should be a nice value.

**ARGUMENTS**

bytes: stack size in bytes.

**DEFAULT**

STACK 4000

## 1.83 Filer.guide/UF\_CF\_CONHANDLER

**CONHANDLER**

-----

**SYNOPSIS**

CONHANDLER "device:", "flags"

**DESCRIPTION**

CONHANDLER sets the conhandler and flags used for Filer's CLI windows. Device is the AmigaDOS device name of the handler. Flags are some additional flags for opening the window like "ALT/", "SMART/" or "NODRAG/". "AUTO/CLOSE/" are added automatically.

**ARGUMENTS**

device: device name of the console handler.  
flags : flags for the console window.

**DEFAULT**

```
CONHANDLER "CON:", ""
```

#### EXAMPLE

```
CONHANDLER "KCON:", "FONTtopaz.11/"
```

## 1.84 Filer.guide/UF\_CF\_EDITOR

### EDITOR

-----

#### SYNOPSIS

```
EDITOR "command"
```

#### DESCRIPTION

EDITOR defines the texteditor used by the "Edit.  
Command is a CLI command which will be called as editor. It  
should contain a "%s" to mark the location where the filename will  
be inserted. The editor should NOT detach from CLI because Filer  
checks the directory for changes after the editor has terminated.

#### ARGUMENTS

command: CLI command to call the texteditor.

#### DEFAULT

```
EDITOR "Ed %s"
```

#### SEE ALSO

Starting external programs

## 1.85 Filer.guide/UF\_CF\_ARCDEF

### ARCDEF

-----

#### SYNOPSIS

```
ARCDEF "pattern", "addcmd", "listcmd", "extractcmd" [,"flags"]
```

#### DESCRIPTION

ARCDEF can be used to add support for any archive file format to  
Filer. ARCDEF is not easy to use, but the supplied config file  
contains configurations for the most popular archive formats. If  
you miss one contact me.

Pattern is an AmigaDOS pattern which matches the filenames of the  
archive files e.g. "#?.lha".

Addcmd is the command for adding files to an archive which must  
contain "%s" twice. The first occurrence one marks the location for  
the filename of the "listfile". This is NOT the file which shall  
be added but a file where all files to be added are listed, one

each line. This method is much faster than adding each file individual. The second "%s" marks the location for the archive's file name. If an archive program does not support creating archives (e.g. "UnZip") use an empty string for addcmd.

Listcmd is the command for listing the contents of an archive. The "%s" marks the location where the filename of the archive should be inserted.

Extractcmd is the command for extracting files from an archive. There must be two "%s" in it: The first one marks the location for the archive's filename, the second one is replaced by the patterns entered by the user.

Flags is an optional parameter for miscellaneous flags. Each character represents a single flag, these ones are supported:

- \* "d":  
If the archive program supports recursive directory scanning itself, you can tell Filer not to do this job. This may have the advantage that empty directories will be archived, too.
- \* "q":  
Some archive programs (e.g. "LhA") need quotes around filenames with spaces some don't (e.g. "LhArc", "LZ" or "Zoo"). This flag turns the generation of quotes on.

#### ARGUMENTS

```
pattern    : AmigaDOS to identify archive files.
addcmd     : CLI command for adding files to an archive.
listcmd    : CLI command for listing files in an archive.
extractcmd : CLI command for extracting files from an archive.
flags      : optional flags.
```

#### EXAMPLE

```
ARCDEF "#?.lha", "LhA -2 -a -e -r -x -i%s a %s", "LhA -a -x v %s",
"LhA -a -m -x x %s %s", "qd"
```

## 1.86 Filer.guide/UF\_CF\_DEFARCEXT

#### DEFARCEXT

#### SYNOPSIS

```
DEFARCEXT "extension"
```

#### DESCRIPTION

DEFARCEXT defines the default extension for filenames of archives created with Filer's "Add function. Specify the extension used by your favourite archiver, here.

#### ARGUMENTS

```
extension: filename extension.
```

---



DEFAULT

DEFARCEXT ""

EXAMPLE

DEFARCEXT ".lha"

## 1.87 Filer.guide/UF\_CF\_IFFDEF

IFFDEF

-----

SYNOPSIS

IFFDEF "type", "command"

DESCRIPTION

IFFDEF can be used to add support for any IFF file format. Type is a IFF identifier like "ILBM", "8SVX" or "ANIM". Command is a CLI command which will be called as a loader. It should contain a "%s" to mark the location where the filename will be inserted. Since version 3.20 Filer performs no special IFF handling any more. "IFFDEF" commands are treated as a short form of the CLASS command.

Filer has internal loaders for pictures ("ILBM") and samples ("8SVX"), but you can define external loaders for these formats, too.

ARGUMENTS

type : IFF identifier.  
command: CLI command to call external loader.

EXAMPLE

IFFDEF "ANIM", "VT %s BUFSIZE 1024 CONT"

SEE ALSO

Starting external programs

## 1.88 Filer.guide/UF\_CF\_TOOL

TOOL

----

SYNOPSIS

TOOL "pattern", "command"

DESCRIPTION

TOOL can be used to add support for any file type. Pattern specifies a file pattern for the file type. Command is a CLI command which will be called as a loader. It should contain a "%s" to mark the location where the filename will be inserted.

---

## ARGUMENTS

pattern: file patterns.  
command: CLI command to call.

## EXAMPLE

TOOL "#?.dvi", "ShowDVI -l %s"

## SEE ALSO

CLASS, REXXCLASS, REXXTOOL, Starting external programs

## 1.89 Filer.guide/UF\_CF\_REXXTOOL

## REXXTOOL

-----

## SYNOPSIS

REXXTOOL "pattern", "command"

## DESCRIPTION

REXXTOOL can be used to add support for any file type. Pattern specifies a pattern for the file type. Command is a ARexx command which will be sent to Filer's ARexx port. It should contain a "%s" to mark the location where the filename will be inserted.

## ARGUMENTS

pattern: file patterns.  
command: ARexx command to call.

## EXAMPLE

REXXTOOL "MOD.#?", "PlayModule %s"

## SEE ALSO

CLASS, REXXCLASS, TOOL

## 1.90 Filer.guide/UF\_CF\_CLASS

## CLASS

-----

## SYNOPSIS

CLASS "pattern", "header", "command"

## DESCRIPTION

CLASS is an extended form of TOOL. Pattern and command were already described there.

Header is a description of the first 32 bytes of the file. Many file formats can't be identified by their filenames, but by some special bytes in their header. You can specify upto 32 bytes and Filer will compare them with the file header. If you only specify

five characters only five will be compared, of course. "?" can be used as a wildcard for any character, no other wildcards are supported.

#### ARGUMENTS

pattern: file patterns.  
header : file header description.  
command: CLI command to call.

#### EXAMPLE

```
CLASS "#?", "GIF8", "VT %s"
CLASS "#?", "??????JFIF", "FJPEG %s"
CLASS "#?.(guide|hyper)", "", "MultiView %f PUBSCREEN %p"
```

#### SEE ALSO

REXXCLASS, REXXTOOL, TOOL, Starting external programs

## 1.91 Filer.guide/UF\_CF\_REXXCLASS

#### REXXCLASS

#### SYNOPSIS

```
REXXCLASS "pattern", "header", "command"
```

#### DESCRIPTION

REXXCLASS is an extended form of REXXTOOL. Pattern and command were already described there.

Header is a description of the first 32 bytes of the file. Many file formats can't be identified by their filenames, but by some special bytes in their header. You can specify up to 32 bytes and Filer will compare them with the file header. If you only specify five characters only five will be compared, of course. "?" can be used as a wildcard for any character, no other wildcards are supported.

#### ARGUMENTS

pattern: file patterns.  
header : file header description.  
command: ARexx command to call.

#### EXAMPLE

```
REXXCLASS "#?.dms", "DMS!", "DMSWrite %s"
```

#### SEE ALSO

CLASS, REXXTOOL, TOOL

## 1.92 Filer.guide/UF\_RX

## Filer's ARexx interface

=====

Filer has a builtin ARexx interface to make it possible to create new functions or to handle file classes in a smarter way.

The name of this port is always "FilerRexx" as only one instance of Filer can run at the same time. Rexx scripts for Filer should have the extension ".filer" to be executed in the normal way.

Many examples for Rexx scripts can be found in the "Rexx" directory of the original distribution archive.

Filer supports the following ARexx commands:

### Basic commands...

VERSION

EXIT

FILERTOFRONT

CD

GETCD

### Commands to invoke builtin functions...

ACTION

ADD

CHGRP

CHOWN

COPY

DELETE

DU

DUPLICATE

EDIT

EXTRACT

FILENOTE

FORMAT

HEXDUMP

LIST

MAKEDIR

MAKELINK

MOVE

PROTECT

QUIT

RELABEL

RENAME

RUN

SEARCH

TYPE

### Commands for controlling the User interface...

ALL

---

BANK  
ICONIFY  
NONE  
PATTERN  
SELECTCACHE

READSOURCEDIR  
READTARGETDIR  
READLEFTDIR  
READRIGHTDIR

ENTERDIR  
OTHERDIR  
PARENTDIR

UP  
PAGEUP  
HOME  
DOWN  
PAGEDOWN  
END  
LEFT  
PAGELEFT  
LEFTMOST  
RIGHT  
PAGERIGHT  
RIGHTMOST

CURSORUP  
CURSORPAGEUP  
CURSORHOME  
CURSORDOWN  
CURSORPAGEDOWN  
CURSOREND

CURSORSELECT  
CURSORSELECTUP  
CURSORSELECTDOWN  
CURSORACTION

Commands for Rexx scripts...

LOCKFILER  
UNLOCKFILER  
PANEL

STATUS  
HISTORY  
CLEARHISTORY

RX  
TEMPFILENAME

ALERTBOX  
QUESTBOX  
FILERREQ  
GETSTRING

---

SETSTRING

EXEC

EXECANDWAIT

FOREACH

GETSCREENNAME

GETSOURCEPATH

GETTARGETPATH

GETLANGUAGE

GETNUMENTRIES

GETNAME

GETINFO

SELECTENTRY

TOGGLEENTRY

UPDATESOURCEDIR

UPDATETARGETDIR

## 1.93 Filer.guide/UF\_RX\_VERSION

VERSION

-----

SYNOPSIS

VERSION

DESCRIPTION

VERSION returns the version information of Filer.

ARGUMENTS

None.

RESULT

Filer's version in format "Filer <VERSION>.<REVISION>".

## 1.94 Filer.guide/UF\_RX\_EXIT

EXIT

----

SYNOPSIS

EXIT

DESCRIPTION

EXIT terminates Filer immediately without any further checks. Use this command only in emergency situations e.g. if an ARexx script which used LOCKFILER terminated before calling UNLOCKFILER. For buttons or keys QUIT should be used.

ARGUMENTS  
None.

RESULT  
None.

## 1.95 Filer.guide/UF\_RX\_FILERTOFRONT

FILERTOFRONT  
-----

SYNOPSIS  
FILERTOFRONT

DESCRIPTION  
FILERTOFRONT brings Filer's screen or window to front. If Filer is in sleep mode it will be reactivated (the screen or window will be reopened).

ARGUMENTS  
None.

RESULT  
None.

## 1.96 Filer.guide/UF\_RX\_CD

CD  
--

SYNOPSIS  
CD DIR/A

DESCRIPTION  
CD changes the current directory of Filer.

ARGUMENTS  
DIR: the new current directory

RESULT  
None.

## 1.97 Filer.guide/UF\_RX\_GETCD

GETCD  
-----

---

## SYNOPSIS

GETCD

## DESCRIPTION

GETCD returns the pathname of the current directory of Filer.

## ARGUMENTS

None.

## RESULT

Current directory of Filer.

## 1.98 Filer.guide/UF\_RX\_ACTION

## ACTION

-----

## SYNOPSIS

ACTION

## DESCRIPTION

ACTION invokes Filer's ACTION function.

## ARGUMENTS

None.

## RESULT

None.

## 1.99 Filer.guide/UF\_RX\_ADD

## ADD

---

## SYNOPSIS

ADD DELETE/S

## DESCRIPTION

ADD invokes Filer's ADD function. If the "DELETE" option is supplied Filer all files successfully added to the archive will be deleted afterwards.

## ARGUMENTS

DELETE: delete files and directories after archiving

## RESULT

None.



## 1.100 Filer.guide/UF\_RX\_CHGRP

CHGRP

-----

SYNOPSIS

CHGRP

DESCRIPTION

CHGRP invokes Filer's Change Group function.

ARGUMENTS

None.

RESULT

None.

## 1.101 Filer.guide/UF\_RX\_CHOWN

CHOWN

-----

SYNOPSIS

CHOWN

DESCRIPTION

CHOWN invokes Filer's Change Owner function.

ARGUMENTS

None.

RESULT

None.

## 1.102 Filer.guide/UF\_RX\_COPY

COPY

----

SYNOPSIS

COPY AS/S

DESCRIPTION

COPY invokes Filer's COPY function. If the "AS" option is supplied Filer will prompt for the name of the copy of each file and directory.

ARGUMENTS

AS: prompt for filenames of copies

RESULT  
None.

### 1.103 Filer.guide/UF\_RX\_DELETE

DELETE  
-----

SYNOPSIS  
DELETE

DESCRIPTION  
DELETE invokes Filer's DELETE function.

ARGUMENTS  
None.

RESULT  
None.

### 1.104 Filer.guide/UF\_RX\_DU

DU  
--

SYNOPSIS  
DU

DESCRIPTION  
DU invokes Filer's Disk Usage function.

ARGUMENTS  
None.

RESULT  
None.

### 1.105 Filer.guide/UF\_RX\_DUPLICATE

Duplicate  
-----

SYNOPSIS  
DUPLICATE

DESCRIPTION  
DUPLICATE invokes Filer's DUPLICATE function.

---

ARGUMENTS  
None.

RESULT  
None.

## 1.106 Filer.guide/UF\_RX\_EDIT

EDIT  
----

SYNOPSIS  
EDIT

DESCRIPTION  
EDIT invokes Filer's EDIT function.

ARGUMENTS  
None.

RESULT  
None.

## 1.107 Filer.guide/UF\_RX\_EXTRACT

EXTRACT  
-----

SYNOPSIS  
EXTRACT

DESCRIPTION  
EXTRACT invokes Filer's EXTRACT function.

ARGUMENTS  
None.

RESULT  
None.

## 1.108 Filer.guide/UF\_RX\_FILENOTE

FILENOTE  
-----

SYNOPSIS

---

FILENOTE

DESCRIPTION

FILENOTE invokes Filer's FILENOTE function.

ARGUMENTS

None.

RESULT

None.

## 1.109 Filer.guide/UF\_RX\_FORMAT

FORMAT

-----

SYNOPSIS

FORMAT

DESCRIPTION

FORMAT invokes Filer's FORMAT function.

ARGUMENTS

None.

RESULT

None.

## 1.110 Filer.guide/UF\_RX\_HEXDUMP

HEXDUMP

-----

SYNOPSIS

HEXDUMP

DESCRIPTION

HEXDUMP invokes Filer's HEXDUMP function.

ARGUMENTS

None.

RESULT

None.

## 1.111 Filer.guide/UF\_RX\_LIST

---

LIST

----

SYNOPSIS

LIST

DESCRIPTION

LIST invokes Filer's LIST function.

ARGUMENTS

None.

RESULT

None.

## 1.112 Filer.guide/UF\_RX\_MAKEDIR

MAKEDIR

-----

SYNOPSIS

MAKEDIR

DESCRIPTION

MAKEDIR invokes Filer's MAKEDIR function.

ARGUMENTS

None.

RESULT

None.

## 1.113 Filer.guide/UF\_RX\_MAKELINK

MAKELINK

-----

SYNOPSIS

MAKELINK

DESCRIPTION

MAKELINK invokes Filer's MAKELINK function.

ARGUMENTS

None.

RESULT

None.

## 1.114 Filer.guide/UF\_RX\_MOVE

MOVE

----

SYNOPSIS

MOVE AS/S

DESCRIPTION

MOVE invokes Filer's MOVE function. If the "AS" option is supplied Filer will prompt for the new filename of each file and directory.

ARGUMENTS

AS: prompt for new filenames

RESULT

None.

## 1.115 Filer.guide/UF\_RX\_PROTECT

PROTECT

-----

SYNOPSIS

PROTECT

DESCRIPTION

PROTECT invokes Filer's PROTECT function.

ARGUMENTS

None.

RESULT

None.

## 1.116 Filer.guide/UF\_RX\_QUIT

QUIT

----

SYNOPSIS

QUIT

DESCRIPTION

QUIT invokes Filer's QUIT function.

ARGUMENTS

None.

RESULT

-----

None.

## 1.117 Filer.guide/UF\_RX\_RELABEL

RELABEL

-----

SYNOPSIS

RELABEL

DESCRIPTION

RELABEL invokes Filer's RELABEL function.

ARGUMENTS

None.

RESULT

None.

## 1.118 Filer.guide/UF\_RX\_RENAME

RENAME

-----

SYNOPSIS

RENAME

DESCRIPTION

RENAME invokes Filer's RENAME function.

ARGUMENTS

None.

RESULT

None.

## 1.119 Filer.guide/UF\_RX\_RUN

RUN

---

SYNOPSIS

RUN

DESCRIPTION

RUN invokes Filer's RUN function.

ARGUMENTS  
None.

RESULT  
None.

## 1.120 Filer.guide/UF\_RX\_SEARCH

SEARCH  
-----

SYNOPSIS  
SEARCH

DESCRIPTION  
SEARCH invokes Filer's SEARCH function.

ARGUMENTS  
None.

RESULT  
None.

## 1.121 Filer.guide/UF\_RX\_TYPE

TYPE  
----

SYNOPSIS  
TYPE

DESCRIPTION  
TYPE invokes Filer's TYPE function.

ARGUMENTS  
None.

RESULT  
None.

## 1.122 Filer.guide/UF\_RX\_ALL

ALL  
---

SYNOPSIS  
ALL

---



## DESCRIPTION

ALL selects all entries in the active directory listview.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "ALL", "+"

## SEE ALSO

NONE

## 1.123 Filer.guide/UF\_RX\_BANK

## BANK

----

## SYNOPSIS

BANK WHICH/A

## DESCRIPTION

BANK switches the active bank of the command panel. With "BANK" you can access up to 256 (twohundredfiftysix) command buttons which should be enough for everyone.  
This command is disabled in the unregistered version.

## ARGUMENTS

WHICH: PREV to go to previous bank  
WHICH: NEXT to go to next bank  
WHICH: 0-7 to go to that bank

## RESULT

The active panel bank.

## CONFIGURATION EXAMPLE

KEY "1", "BANK 0"  
KEY "2", "BANK 1"  
BUTTON 0, 7, "->", "BANK NEXT"  
BUTTON 1, 7, "<-", "BANK PREV"

## 1.124 Filer.guide/UF\_RX\_ICONIFY

## ICONIFY

-----

## DESCRIPTION

ICONIFY

---

## SYNOPSIS

ICONIFY puts Filer in an iconified state. Filer frees most of it's resources and waits for reactivation via it's ARexx interface. If Filer is waked up it will reload the directories that were in the directory listviews before Filer was iconified.

## ARGUMENTS

None.

## RESULT

None.

## 1.125 Filer.guide/UF\_RX\_NONE

## NONE

----

## SYNOPSIS

NONE

## DESCRIPTION

NONE deselects all entries in the active directory listview.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "NONE", "-"

## SEE ALSO

ALL

## 1.126 Filer.guide/UF\_RX\_PATTERN

## PATTERN

-----

## SYNOPSIS

PATTERN

## DESCRIPTION

PATTERN opens the pattern matching requester for the active directory listview.

## ARGUMENTS

None.

---

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "HELP", "PATTERN"

## SEE ALSO

Pattern Gadget

## 1.127 Filer.guide/UF\_RX\_SELECTCACHE

## SELECTCACHE

-----

## SYNOPSIS

SELECTCACHE

## DESCRIPTION

SELECTCACHE opens the cache selection requester for the active directory listview.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "SHIFT ?", "SELECTCACHE"

## SEE ALSO

Cache Gadget

## 1.128 Filer.guide/UF\_RX\_READSOURCEDIR

## READSOURCEDIR

-----

## SYNOPSIS

READSOURCEDIR DIR/A

## DESCRIPTION

READSOURCEDIR reads the specified directory in the active directory listview. If the directory is read successful the complete pathname is returned otherwise nothing.

## ARGUMENTS

DIR: Pathname of the directory to read.

## RESULT

Full pathname of the directory

## CONFIGURATION EXAMPLE

KEY "F1", "READSOURCEDIR SYS:"

## SEE ALSO

READTARGETDIR, READLEFTDIR, READRIGHTDIR,

## 1.129 Filer.guide/UF\_RX\_READTARGETDIR

## READTARGETDIR

-----

## SYNOPSIS

READTARGETDIR DIR/A

## DESCRIPTION

READTARGETDIR reads the specified directory in the inactive directory listview. If the directory is read successful the complete pathname is returned otherwise nothing.

## ARGUMENTS

DIR: Pathname of the directory to read

## RESULT

Full pathname of the directory.

## CONFIGURATION EXAMPLE

KEY "SHIFT F1", "READTARGETDIR SYS:"

## SEE ALSO

READSOURCEDIR, READLEFTDIR, READRIGHTDIR,

## 1.130 Filer.guide/UF\_RX\_READLEFTDIR

## READLEFTDIR

-----

## SYNOPSIS

READLEFTDIR DIR/A

## DESCRIPTION

READLEFTDIR reads the specified directory in the left directory listview. If the directory is read successful the complete pathname is returned otherwise nothing.

## ARGUMENTS

DIR: Pathname of the directory to read

---

## RESULT

Full pathname of the directory.

## CONFIGURATION EXAMPLE

BUTTON 0,0,"DF0:","READLEFTDIR DF0:"

## SEE ALSO

READRIGHTDIR, READSOURCEDIR, READTARGETDIR,

## 1.131 Filer.guide/UF\_RX\_READRIGHTDIR

## READRIGHTDIR

-----

## SYNOPSIS

READRIGHTDIR DIR/A

## DESCRIPTION

READRIGHTDIR reads the specified directory in the right directory listview. If the directory is read successful the complete pathname is returned otherwise nothing.

## ARGUMENTS

DIR: Pathname of the directory to read

## RESULT

Full pathname of the directory.

## CONFIGURATION EXAMPLE

BUTTON 0,4,"DF0:","READRIGHTDIR DF0:"

## SEE ALSO

READLEFTDIR, READSOURCEDIR, READTARGETDIR,

## 1.132 Filer.guide/UF\_RX\_ENTERDIR

## ENTERDIR

-----

## SYNOPSIS

ENTERDIR

## DESCRIPTION

ENTERDIR activates the pathname gadget of the active directory listview so a new directory pathname can be entered.

## ARGUMENTS

None.

## RESULT

None.

#### CONFIGURATION EXAMPLE

KEY "RETURN", "ENTERDIR"

#### SEE ALSO

Path Gadget

## 1.133 Filer.guide/UF\_RX\_OTHERDIR

### OTHERDIR

-----

#### SYNOPSIS

OTHERDIR

#### DESCRIPTION

OTHERDIR activates the currently inactive directory listview.

#### ARGUMENTS

None.

#### RESULT

None.

#### CONFIGURATION EXAMPLE

KEY "TAB", "OTHERDIR"

## 1.134 Filer.guide/UF\_RX\_PARENTDIR

### PARENTDIR

-----

#### SYNOPSIS

PARENTDIR

#### DESCRIPTION

PARENTDIR loads the parent directory of the active directory listview into it. If there is no parent directory nothing is done.

#### ARGUMENTS

None.

#### RESULT

None.

#### CONFIGURATION EXAMPLE

KEY "SHIFT /", "PARENTDIR"

---

## 1.135 Filer.guide/UF\_RX\_UP

UP  
--

SYNOPSIS  
UP

DESCRIPTION  
UP moves the visible region of the active directory listview one step up. The cursor is only moved if it's on the lower end of the visible region.

ARGUMENTS  
None.

RESULT  
None.

CONFIGURATION EXAMPLE  
KEY "NUMPAD 8", "UP"

SEE ALSO  
DOWN, CURSORUP

## 1.136 Filer.guide/UF\_RX\_PAGEUP

PAGEUP  
-----

SYNOPSIS  
PAGEUP

DESCRIPTION  
PAGEUP moves the visible region of the active directory listview one page up. If the cursor gets out of the screen it's moved to the lower end of the visible region.

ARGUMENTS  
None.

RESULT  
None.

CONFIGURATION EXAMPLE  
KEY "NUMPAD 9", "PAGEUP"

SEE ALSO  
PAGEDOWN, CURSORPAGEUP

---

## 1.137 Filer.guide/UF\_RX\_HOME

HOME

----

SYNOPSIS

HOME

DESCRIPTION

HOME moves the visible region of the active directory listview to the top. If the cursor gets out of the screen it's moved to the lower end of the visible region.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "NUMPAD 7", "HOME"

SEE ALSO

END, CURSORHOME

## 1.138 Filer.guide/UF\_RX\_DOWN

DOWN

----

SYNOPSIS

DOWN

DESCRIPTION

DOWN moves the visible region of the active directory listview one step down. The cursor is only moved if it's on the upper end of the visible region.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "NUMPAD 2", "DOWN"

SEE ALSO

UP, CURSORDOWN

---



## 1.139 Filer.guide/UF\_RX\_PAGEDOWN

PAGEDOWN

-----

SYNOPSIS

PAGEDOWN

DESCRIPTION

PAGEDOWN moves the visible region of the active directory listview one page down. If the cursor gets out of the screen it's moved to the upper end of the visible region.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "NUMPAD 3", "PAGEDOWN"

SEE ALSO

PAGEUP, CURSORPAGEDOWN

## 1.140 Filer.guide/UF\_RX\_END

END

---

SYNOPSIS

END

DESCRIPTION

END moves the visible region of the active directory listview to the bottom. If the cursor gets out of the screen it's moved to the upper end of the visible region.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "NUMPAD 1", "HOME"

SEE ALSO

HOME, CURSOREND

## 1.141 Filer.guide/UF\_RX\_LEFT

LEFT

----

SYNOPSIS

LEFT

DESCRIPTION

LEFT moves the visible region of the active directory listview one step to the left.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "CURSOR\_LEFT", "LEFT"

SEE ALSO

RIGHT

## 1.142 Filer.guide/UF\_RX\_PAGELEFT

PAGELEFT

-----

SYNOPSIS

PAGELEFT

DESCRIPTION

PAGELEFT moves the visible region of the active directory listview one page to the left.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "SHIFT CURSOR\_LEFT", "PAGELEFT"

SEE ALSO

PAGERIGHT

## 1.143 Filer.guide/UF\_RX\_LEFTMOST

LEFTMOST

-----

SYNOPSIS

LEFTMOST

DESCRIPTION

LEFTMOST moves the visible region of the active directory listview to its leftmost position.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "CONTROL\_CURSOR\_LEFT", "LEFTMOST"

SEE ALSO

RIGHTMOST

## 1.144 Filer.guide/UF\_RX\_RIGHT

RIGHT

-----

SYNOPSIS

RIGHT

DESCRIPTION

RIGHT moves the visible region of the active directory listview one step to the right.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "CURSOR\_RIGHT", "RIGHT"

SEE ALSO

LEFT

## 1.145 Filer.guide/UF\_RX\_PAGERIGHT

PAGERIGHT

---

-----

#### SYNOPSIS

PAGERIGHT

#### DESCRIPTION

PAGERIGHT moves the visible region of the active directory listview one page to the right.

#### ARGUMENTS

None.

#### RESULT

None.

#### CONFIGURATION EXAMPLE

KEY "SHIFT CURSOR\_RIGHT", "PAGERIGHT"

#### SEE ALSO

PAGERLEFT

## 1.146 Filer.guide/UF\_RX\_RIGHTMOST

#### RIGHTMOST

-----

#### SYNOPSIS

RIGHTMOST

#### DESCRIPTION

RIGHTMOST moves the visible region of the active directory listview to its rightmost position.

#### ARGUMENTS

None.

#### RESULT

None.

#### CONFIGURATION EXAMPLE

KEY "CONTROL CURSOR\_RIGHT", "RIGHTMOST"

#### SEE ALSO

LEFTMOST

## 1.147 Filer.guide/UF\_RX\_CURSORUP

#### CURSORUP

-----

---

## SYNOPSIS

CURSORUP

## DESCRIPTION

CURSORUP moves the cursor one step up. If the cursor is at the top of the visible region of active directory listview Filer scrolls one step up.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "CURSOR\_UP", "CURSORUP"

## SEE ALSO

CURSORDOWN, UP

## 1.148 Filer.guide/UF\_RX\_CURSORPAGEUP

## CURSORPAGEUP

-----

## SYNOPSIS

CURSORPAGEUP

## DESCRIPTION

CURSORPAGEUP moves the cursor to the top of the visible region of active directory listview. If it is already there Filer scrolls one page up.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "SHIFT CURSOR\_UP", "CURSORPAGEUP"

## SEE ALSO

CURSORPAGEDOWN, PAGEUP

## 1.149 Filer.guide/UF\_RX\_CURSORHOME

## CURSORHOME

-----

---

## SYNOPSIS

CURSORHOME

## DESCRIPTION

CURSORHOME moves the cursor and the top of the visible region to the first entry.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "CONTROL\_CURSOR\_UP", "CURSORHOME"

## SEE ALSO

CURSOREND, HOME

## 1.150 Filer.guide/UF\_RX\_CURSORDOWN

## CURSORDOWN

-----

## SYNOPSIS

CURSORDOWN

## DESCRIPTION

CURSORDOWN moves the cursor one step down. If the cursor is at the bottom of the visible region of active directory listview Filer scrolls one step down.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "CURSOR\_DOWN", "CURSORDOWN"

## SEE ALSO

CURSORUP, DOWN

## 1.151 Filer.guide/UF\_RX\_CURSORPAGEDOWN

## CURSORPAGEDOWN

-----

## SYNOPSIS

CURSORMAGEDOWN

DESCRIPTION

CURSORMAGEDOWN moves the cursor to the bottom of the visible region of active directory listview. If it is already there Filer scrolls one page down.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "SHIFT CURSOR\_DOWN", "CURSORMAGEDOWN"

SEE ALSO

CURSORMAGEDUP, MAGEDOWN

## 1.152 Filer.guide/UF\_RX\_CURSOREND

CURSOREND

-----

SYNOPSIS

CURSOREND

DESCRIPTION

CURSOREND moves the cursor and the bottom of the visible region to the last entry.

ARGUMENTS

None.

RESULT

None.

CONFIGURATION EXAMPLE

KEY "CONTROL CURSOR\_DOWN", "CURSOREND"

SEE ALSO

CURSORMAGEDUP, END

## 1.153 Filer.guide/UF\_RX\_CURSORSELECT

CURSORSELECT

-----

SYNOPSIS

CURSORSELECT

## DESCRIPTION

CURSORSELECT toggles the entry under the cursor.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "SPACE", "CURSOSELECT"

## SEE ALSO

CURSORSELECTDOWN, CURSORSELECTUP

## 1.154 Filer.guide/UF\_RX\_CURSORSELECTDOWN

## CURSORSELECTDOWN

## SYNOPSIS

CURSORSELECTDOWN

## DESCRIPTION

CURSORSELECTDOWN toggles the entry under the cursor and moves the cursor one step down. If the cursor is at the bottom of the visible region of active directory listview Filer scrolls one step down.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "SPACE", "CURSOSELECTDOWN"

## SEE ALSO

CURSORSELECT, CURSORSELECTUP

## 1.155 Filer.guide/UF\_RX\_CURSORSELECTUP

## CURSORSELECTUP

## SYNOPSIS

CURSORSELECTUP



## DESCRIPTION

CURSORSELECTUP toggles the entry under the cursor and moves the cursor one step up. If the cursor is at the top of the visible region of active directory listview Filer scrolls one step up.

## ARGUMENTS

None.

## RESULT

None.

## CONFIGURATION EXAMPLE

KEY "SPACE", "CURSOSELECTUP"

## SEE ALSO

CURSORSELECT, CURSORSELECTDOWN

## 1.156 Filer.guide/UF\_RX\_CURSORACTION

## CURSORACTION

-----

## SYNOPSIS

CURSORACTION

## DESCRIPTION

CURSORACTION performs an auto-action on the entry under the cursor.

## ARGUMENTS

None.

## RESULT

None.

## 1.157 Filer.guide/UF\_RX\_LOCKFILER

## LOCKFILER

-----

## SYNOPSIS

LOCKFILER

## DESCRIPTION

LOCKFILER obtains a lock for Filer. While you have got this lock Filer can neither be quit the normal way nor get locked again. This mechanism is very useful for scripts.

## ARGUMENTS

None.

---

## RESULT

A lock needed for UNLOCKFILER or nothing if Filer is already locked. If this happens a Rexx script should terminate.

## SEE ALSO

EXIT, UNLOCKFILER

## 1.158 Filer.guide/UF\_RX\_UNLOCKFILER

## UNLOCKFILER

-----

## SYNOPSIS

UNLOCKFILER KEY/N/A

## DESCRIPTION

UNLOCKFILER unlocks Filer after it has been locked with LOCKFILER. Without this call Filer can NOT be terminated the normal way, so ensure that your Rexx scripts always call UNLOCKFILER. If a script crashes use EXIT.

## ARGUMENTS

KEY: A lock obtained by LOCKFILER

## RESULT

None.

## SEE ALSO

EXIT, LOCKFILER

## 1.159 Filer.guide/UF\_RX\_PANEL

## PANEL

-----

## SYNOPSIS

PANEL STATUS/A

## DESCRIPTION

PANEL turns Filer's command panel on or off. If it's turned off the window will get a busy pointer and no inputs are accepted until the panel is turned on again. ARexx scripts should always use this command to prevent the user from doing nasty things while a script is running.

## ARGUMENTS

STATUS: either ON or OFF.

## RESULT

None.

---

## 1.160 Filer.guide/UF\_RX\_STATUS

STATUS

-----

SYNOPSIS

STATUS MSG/F

DESCRIPTION

STATUS changes the contents of Filer's status bar to the supplied message. The message is NOT added to the history buffer.

ARGUMENTS

MSG: message text for status bar.

RESULT

None.

SEE ALSO

Status Bar, HISTORY, CLEARHISTORY

## 1.161 Filer.guide/UF\_RX\_HISTORY

HISTORY

-----

SYNOPSIS

HISTORY MSG/F

DESCRIPTION

HISTORY changes the contents of Filer's status bar to the supplied message. The message is added to the history buffer.

ARGUMENTS

MSG: message text for status bar.

RESULT

None.

SEE ALSO

Status Bar, STATUS, CLEARHISTORY

## 1.162 Filer.guide/UF\_RX\_CLEARHISTORY

## CLEARHISTORY

-----

### SYNOPSIS

CLEARHISTORY

### DESCRIPTION

CLEARHISTORY clears Filer's history buffer.

### ARGUMENTS

None.

### RESULT

None.

### SEE ALSO

Status Bar, HISTORY, STATUS

## 1.163 Filer.guide/UF\_RX\_RX

### RX

--

### SYNOPSIS

RX CMD/F/A

### DESCRIPTION

RX executes a ARexx command in Filer's shell window. This way a script or CLI commands called by it can use Filer's window for input and output. To start a Rexx script you can supply the shortform "ScriptName" instead of "Rexx:ScriptName.filer".

### ARGUMENTS

CMD: ARexx command to execute

### RESULT

None.

## 1.164 Filer.guide/UF\_RX\_TEMPFILENAME

### TEMPFILENAME

-----

### SYNOPSIS

TEMPFILENAME

### DESCRIPTION

TEMPFILENAME generates a unique name for a temporary file. These names are unique for the lifetime of Filer so this command can be

used mutiple times in a ARexx script. The filename alway points to the directory "T:".

**ARGUMENTS**

None.

**RESULT**

An unique name for a temporary file.

## 1.165 Filer.guide/UF\_RX\_ALERTBOX

**ALERTBOX**

-----

**SYNOPSIS**

ALERTBOX MSG/F/A

**DESCRIPTION**

ALERTBOX opens a requester with a single button and displays the supplied message. Multiple lines in the message text may be seperated by "|" characters.

**ARGUMENTS**

MSG: message text for requester.

**RESULT**

None.

**SEE ALSO**

QUESTBOX, FILERREQ

## 1.166 Filer.guide/UF\_RX\_QUESTBOX

**QUESTBOX**

-----

**SYNOPSIS**

QUESTBOX MSG/F/A

**DESCRIPTION**

QUESTBOX opens a requester with two buttons labeled "Yes" and "No" and displays the supplied message. Multiple lines in the message text may be separated by "|" characters. If the user answers "positive" true (1) is returned else false (0) is returned.

**ARGUMENTS**

MSG: message text for requester.

**RESULT**

User's choice.

---

SEE ALSO

ALERTBOX, FILERREQ

## 1.167 Filer.guide/UF\_RX\_FILERREQ

FILERREQ

-----

SYNOPSIS

FILERREQ TITLE/A,MSG/A,GADGETS/M/A

DESCRIPTION

FILERREQ is the most general command to open a Filer requester. You must specify the window title, the message text and labels for one or more gadgets. Multiple lines in the message text may be separated by "|" characters. You can use underscores in the gadget labels to mark the following character as hotkey for that button. The leftmost button has always RETURN and SPACE as additional hotkeys, the rightmost one ESC. If the user selects the leftmost gadget 0 is returned, the next one returns 1, ..., the rightmost one returns the number of gadgets minus one.

ARGUMENTS

TITLE : requester's window title  
MSG : message text for the requester  
GADGETS: labels for one or more gadgets

RESULT

User's choice.

SEE ALSO

ALERTBOX, QUESTBOX

## 1.168 Filer.guide/UF\_RX\_GETSTRING

GETSTRING

-----

SYNOPSIS

GETSTRING MSG/F/A

DESCRIPTION

GETSTRING opens a string requester with two buttons labeled "Okay" and "Cancel", displays the supplied message and ask for string. Multiple lines in the message text may be separated by "|" characters. If the user completes the input by pressing RETURN or clicks on the "Okay" gadget the input string is returned else nothing is returned. The initial contents of the string gadget may be set with SETSTRING.

---

## ARGUMENTS

MSG: message text for requester.

## RESULT

User's input.

## SEE ALSO

SETSTRING

## 1.169 Filer.guide/UF\_RX\_SETSTRING

## SETSTRING

-----

## SYNOPSIS

SETSTRING STRING/F

## DESCRIPTION

SETSTRING sets the initial contents for the string gadget in the requester opened by GETSTRING.

## ARGUMENTS

STRING: contents for the string gadget.

## RESULT

None.

## SEE ALSO

GETSTRING

## 1.170 Filer.guide/UF\_RX\_EXEC

## EXEC

----

## SYNOPSIS

EXEC CMD/F/A

## DESCRIPTION

EXEC opens a CLI window on Filer's screen and performs the supplied CLI command.

## ARGUMENTS

CMD: CLI command to be executed

## RESULT

None.

## SEE ALSO

EXECANDWAIT, FOREACH, Starting external programs

## 1.171 Filer.guide/UF\_RX\_EXECANDWAIT

EXECANDWAIT

-----

### SYNOPSIS

EXECANDWAIT CMD/F/A

### DESCRIPTION

EXECANDWAIT opens a CLI window on Filer's screen and performs the supplied CLI command. If the command does any output to the window the user will be requested to press RETURN after the command has completed.

### ARGUMENTS

CMD: CLI command to be executed

### RESULT

None.

### SEE ALSO

EXEC, FOREACH, Starting external programs

## 1.172 Filer.guide/UF\_RX\_FOREACH

FOREACH

-----

### SYNOPSIS

FOREACH COMMAND/A,DIRS/S,WAIT/S

### DESCRIPTION

FOREACH is a simple method to integrate external programs into Filer. The supplied CLI command will be called for each selected file and if the "DIRS" option is set for each selected directory, too. It should contain a "%s" to mark the location where the filename will be inserted. Don't forget the quotes around the command.

### ARGUMENTS

COMMAND: CLI command to call foreach file

DIR : process also directories

WAIT : wait for an acknowledgement after the last file was processed

### RESULT

None.

### CONFIGURATION EXAMPLE

---



```
    BUTTON ..., ..., "GZip", "FOREACH \"gzip -9 %s\""
```

SEE ALSO

EXEC, EXECANDWAIT, Starting external programs

### 1.173 Filer.guide/UF\_RX\_GETSCREENNAME

GETSCREENNAME

-----

SYNOPSIS

GETSCREENNAME

DESCRIPTION

GETSCREENNAME returns the name of the public screen on which Filer's window was opened. This screen may be either a screen opened by Filer, the Workbench screen or any other public screen. This name can be used to open own windows on Filer's screen.

ARGUMENTS

None.

RESULT

Name of the public screen on which Filer's window was opened.

### 1.174 Filer.guide/UF\_RX\_GETSOURCEPATH

GETSOURCEPATH

-----

SYNOPSIS

GETSOURCEPATH

DESCRIPTION

GETSOURCEPATH returns the pathname of the directory in the active directory listview. If no directory is loaded nothing is returned.

ARGUMENTS

None.

RESULT

Pathname of the directory in the active directory listview.

SEE ALSO

GETTARGETPATH

### 1.175 Filer.guide/UF\_RX\_GETTARGETPATH

---

GETTARGETPATH

-----

SYNOPSIS

GETTARGETPATH

DESCRIPTION

GETTARGETPATH returns the pathname of the directory in the active directory listview. If no directory is loaded nothing is returned.

ARGUMENTS

None.

RESULT

Pathname of the directory in the inactive directory listview.

SEE ALSO

GETSOURCEPATH

## 1.176 Filer.guide/UF\_RX\_GETLANGUAGE

GETLANGUAGE

-----

SYNOPSIS

GETLANGUAGE

DESCRIPTION

GETLANGUAGE returns the language of the catalog currently used by Filer.

ARGUMENTS

None.

RESULT

The current language.

## 1.177 Filer.guide/UF\_RX\_GETNUMENTRIES

GETNUMENTRIES

-----

SYNOPSIS

GETNUMENTRIES

DESCRIPTION

GETNUMENTRIES returns the number of the entries of the directory shown in the active directory listview.

## ARGUMENTS

None.

## RESULT

Number of entries in the active directory listview.

## 1.178 Filer.guide/UF\_RX\_GETNAME

## GETNAME

-----

## SYNOPSIS

GETNAME INDEX/N/A

## DESCRIPTION

GETNAME returns the filename of an entry in the active directory listview. The name is preceded by a single character which is either a "D" for a directory or a "F" for file. If this character is lowercase the entry is selected.

## ARGUMENTS

INDEX: index of an entry [1-number of entries].

## RESULT

Filename with special prefix.

## EXAMPLES

Results of calls of GETNAME could look like this:

\* "fFoo": selected file called "Foo"

\* "Dbar": not selected directory called "bar"

## SEE ALSO

GETINFO

## 1.179 Filer.guide/UF\_RX\_GETINFO

## GETINFO

-----

## SYNOPSIS

GETINFO INDEX/N/A,FORMAT

## DESCRIPTION

GETINFO returns information about an entry in the active directory listview.

## ARGUMENTS

INDEX : index of an entry [1-number of entries]. FORMAT: format of the result.

---

## RESULT

Information in special format.

## EXAMPLES

Results of calls of GETINFO could look like this:

```
*          ---RWED DI 20-Jan-94 09:38:24

*          213 -S-RW-D FI 19-Dez-93 18:10:32
```

## SEE ALSO

GETNAME

## 1.180 Filer.guide/UF\_RX\_SELECTENTRY

## SELECTENTRY

-----

## SYNOPSIS

SELECTENTRY NAME/A

## DESCRIPTION

SELECTENTRY selects the specified file or directory if it is present in the active directory listview. The command is mainly useful for Filer's commodity.

## ARGUMENTS

NAME: Filename of a file or directory

## RESULT

None.

## 1.181 Filer.guide/UF\_RX\_TOGGLEENTRY

## TOGGLEENTRY

-----

## SYNOPSIS

TOGGLEENTRY INDEX/N/A

## DESCRIPTION

TOGGLEENTRY toggles the state of the entry with the supplied index in the active directory listview.

## ARGUMENTS

INDEX: index of an entry [1-number of entries].

## RESULT

None.

---

## 1.182 Filer.guide/UF\_RX\_UPDATESOURCEDIR

UPDATESOURCEDIR

### SYNOPSIS

UPDATESOURCEDIR

### DESCRIPTION

UPDATESOURCEDIR checks whether the directory in the active listview has been changed since it has been read from disk. If this is indeed true the directory is reread. This command may be used from ARExx scripts which modify the contents of the directory e.g. by calling a CLI command.

### ARGUMENTS

None.

### RESULT

None.

### SEE ALSO

UPDATETARGETDIR

## 1.183 Filer.guide/UF\_RX\_UPDATETARGETDIR

UPDATETARGETDIR

### SYNOPSIS

UPDATETARGETDIR

### DESCRIPTION

UPDATETARGETDIR checks whether the directory in the inactive listview has been changed since it has been read from disk. If this is indeed true the directory is reread. This command may be used from ARExx scripts which modify the contents of the directory e.g. by calling a CLI command.

### ARGUMENTS

None.

### RESULT

None.

### SEE ALSO

UPDATESOURCEDIR

## 1.184 Filer.guide/UF\_EXT

---

## Starting external programs

=====

You can use external programs to extend Filer's possibilities. The can be used to edit or view textfiles, to automatically handle files or to create new functions via the FOREACH ARExx command.

These programs will always be invoked in a CLI environment. If a custom shell e.g. WShell is installed it will be used instead of the standard system shell. The stack, path and console handler used for this shell can be defined in the configuration file.

You usually supply a synopsis for calling the external programm. You can use these special strings in this synopsis:

- \* %%: the percent sign
- \* %s: the name of the involved file
- \* %f: the full path of the involved file
- \* %p: the name of Filer's PubScreen
- \* %S: the full path of the source directory
- \* %T: the full path of the target directory

If you refer to an undefined directory an error message is displayed.

## 1.185 Filer.guide/UF\_CX

### Filer's Commodity

=====

Filer comes with a commodity called "FilerCom". This commodity can be used to start Filer via a HotKey or an AppIcon on the Workbench. The advantage of an external program for this purpose is that you don't have to keep Filer in memory.

Arguments	Arguments supported by FilerCom.
HotKey	Starting Filer via HotKey.
AppIcon	Using the Application Icon.

## 1.186 Filer.guide/UF\_CX\_ARGS

### Arguments

-----

FilerCom supports these arguments:

\* PATH=path

Path of the main program, default: "SYS:Tools/Filer"

\* AUTOREXXCMD=cmd

ARexx command for AppIcon, default: no command

\* ICON=ON|OFF

Turn AppIcon on or off, default: OFF

\* ICONXPOS=xpos

x-position of the AppIcon, default: Workbench's choice

\* ICONYPOS=ypos

y-position of the AppIcon, default: Workbench's choice

\* ICONNAME=name

Name of the AppIcon, default: "Filer"

\* ICONFILE=path

Name of the icon file for the AppIcon, default: "SYS:Tools/Filer"

\* CX\_PRIORITY=priority

commodity priority

\* HOTKEY=description

hotkey for invoking Filer, default "LCOMMAND f"

If you start FilerCom from the Workbench (e.g. in the "WBStartup" drawer) put these arguments in the tooltypes of its icon. If you start it from CLI supply them as command line arguments including the "=".

## 1.187 Filer.guide/UF\_CX\_HOTKEY

Starting Filer via HotKey

-----

If FilerCom is installed you just have to press the HotKey and Filer will be loaded. This can take some time especially on slow system, so be patient. If Filer is already in memory FilerCom will bring Filer's screen to front.

---

## 1.188 Filer.guide/UF\_CX\_APPICON

Using the Application Icon

FilerCom's AppIcon can be used in different ways:

- \* Doubleclick on the AppIcon

After doubleclicking on the AppIcon Filer will be loaded. If Filer is already in memory FilerCom will bring Filer's screen to front.

- \* Drop one or two drawer icons on the AppIcon

If you drop a drawer icon on FilerCom's AppIcon the directory will be read into Filer. If you also drop a second drawer icon this directory will be read, too, but the directory listview with the first directory will be reactivated.

- \* Drop tool or project icons on the AppIcon

If you drop icons of files on FilerCom's AppIcon the directory which contains these files will be read into Filer and the files will be selected. If you have specified a ARexx command as argument this ARexx command will be executed. A good choice for this command is ACTION.

## 1.189 Filer.guide/OT\_UPD

Other topics

\*\*\*\*\*

Updates

=====

Whenever a new release of Filer gets released, I will post some information in the appropriate newsgroups of some electronic networks. The new archive will soon be available on many bulletin boards and on all AmiNet FTP servers. Major releases will also come with some PD disks.

As mentioned above, registered users will neither need a new keyfile nor a special personalized program version. They can use all new features immediately.

Owners of Filer version 2.0x or 3.0x have to send a letter (including postage), a disk, their address and 10,- DM to me to receive a keyfile. There is no update option for the version 1.2 released on the Amiga Power Disk 9.



## 1.190 Filer.guide/OT\_SUP

### Support

=====

If you have some questions, comments, suggestions or even flames, please feel free to contact me at one of the following addresses. If you send your letter via e-mail, there's a good chance for getting a quick reply.

Snailmail: Matthias Scheler  
Schützenstraße 18  
33178 Borchten  
FR Germany

Phone: +49-5251-399031

E-Mail: filer@lyssa.owl.de  
or: Matthias Scheler 2:243/6350.18

## 1.191 Filer.guide/OT\_HIST

### History

=====

- \* 0.1 - 1.1  
not released
  - \* 1.2  
released on Amiga Power Disk 9
  - \* 1.3 - 1.4  
not released
  - \* 2.00 - 2.07  
released on Fish Disk 562 as Shareware, new features:
    - \* requires OS 2.04 or newer
    - \* ASCII configuration file
    - \* support for other IFF formats
    - \* class support
    - \* archive file support
  - \* 3.00 - 3.05  
sold to IPV but never released by them, new features:
    - \* ARexx interface
    - \* complete GUI made with GadTools
    - \* user configurable main panel
-

- \* screenmode support
  - \* extended class support
  - \* new RTG-compatible textviewer
  - \* 3.10
    - Shareware again, new features:
      - \* supports now window mode on any PubScreen
      - \* font sensitive (incl. complete new preference requesters)
      - \* keyboard support
      - \* ARexx interface extended
      - \* textviewer supports ANSI control characters, less memory required
    - and many more ...
  - \* 3.11
    - \* fixed MungWall hit in the audio player.
    - \* supports samples as large as largest block of any free memory.
    - \* increased maximum header size for CLASS and REXXCLASS.
  - \* 3.12
    - \* fixed small bug in new audio player
    - \* screen switching without stoping the audio player is now possible
    - \* some small improvements
  - \* 3.13
    - \* sizegadget of the main window now at the bottom border
    - \* AUTOREQUESTER option now off per default
    - \* added work arround for one of the ExAll() bugs under V37
    - \* improved memory management
  - \* 3.14
    - \* no more crashes with BAD 8SVX-files
    - \* location of the configuration file can be setup
    - \* if fonts are not configured system default fonts are used
  - \* 3.15
    - \* improved audio player
      - \* uses "CHAN" chunk to identify stereo samples
-

- \* Loop can be switch on and off via config option
  - \* audio Filter can be switch on and off via config option
  - \* new config option SELECTICON
  - \* fixed some small bugs
  - \* Filer Commodity 1.20
    - \* FilerCom can be used to restart/iconify Filer by using the "Show"/ "Hide" gadget in the "Exchange" program.
  - \* Filer does not detach from CLI anymore. This feature was somehow hacked and it confused SegTracker. And in post 1.3 times there are other ways to start programs in the background.
  - \* 3.20
    - \* finally localized
    - \* Improved ARexx interface
      - \* ReadArgs() style argument parsing
      - \* external programs or scripts launched from Filer can use the ARexx interface now, no more lookups
      - \* two powerful new commands FILERREQ and FOREACH
      - \* STARTUP and SHUTDOWN command
    - \* added (optional) invisible parent gadget at left and right border like in MagicFileRequester (controlled with the BORDERPARENTDIR configuration flag)
    - \* Add & Delete
    - \* Copy As
    - \* Move As
    - \* optional fancy listview updating
    - \* optional overwrite protection while copying files
    - \* optional automatic diskchange recognition
    - \* some small bug fixes and improvements
    - \* Filer Commodity 1.30
      - \* adapted to the new ARexx interface
  - \* 3.21
    - \* menu item for snapshotting the main window
    - \* fixed some bugs especially in the display update
  - \* 3.22
-

- \* some small bug fixes
- \* improved "Hide" requester
- \* improved ARexx command GETINFO
- \* Envoy users and groups are shown by name.
- \* new funktionen Change Owner and Change Group
- \* Filer Commodity 1.31
  - \* Version number in the Commodity description was wrong.
- \* 3.23
  - \* some small bug fixes

## 1.192 Filer.guide/OT\_FUT

Future  
=====

Here are some ideas for future versions of Filer:

- \* Filename completion and history in the path gadgets
- \* XPK-Support
- \* User defined menus
- \* better IFF support (24 Bit ILBM files, large 8SVX files)

Important:

There is absolutely NO guarantee that these features will ever be implemented. So don't be disappointed, if they aren't in the next version.

## 1.193 Filer.guide/OT\_ACK

Acknowledgments  
=====

Thanks must go to:

- Mario Kemper

for his never ending feature requests, criticism and bug reports  
and especially the fun we had with the "User incompatible problem"

- Markus Stipp

---

for lots of bug reports and helping me to make Filer style guide proof

- Christian Berger

for being a "normal" user which helped me to make Filer more user friendly and intuitive to use

- Ralph Schmidt

for his realtime assembler Barfly which is used to translate the assembler parts of Filer

- Michael "mick" Hohman

for collecting nice features from other file managers and telling me about them in his special way

- Michael Illgner

for his strange feature requests

- Michael "billy" Böhnisch

for his nice collection of ARexx scripts and finding errors in this manual

- Michael Sinz

for his fantastic debugging tool Enforcer which helped me to find lots of nasty bugs in my program (ask me for a keyfile and you'll get one)

- Stefan Stuntz and Stefan Becker

for their manuals for MUI respectively ToolManager which were good examples for this one

And of course to all the other Beta testers and registered users.

---