

zDir

Finn Nielsen <zznyyd@diku.dk>

Copyright © CopyrightÂ©1997 ZzNyYd! Design.

COLLABORATORS

	<i>TITLE :</i> zDir		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Finn Nielsen <zznyyd@diku.dk>	August 6, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	zDir	1
1.1	zDir V2.0 (16.8.97) by Finn Nielsen - ©1997 ZzNyYd! Design.	1
1.2	zDir V2.0 - Introduction	2
1.3	zDir V2.0 - Background	2
1.4	zDir V2.0 - Features	3
1.5	zDir V2.0 - Disclaimer	4
1.6	zDir V2.0 - Distribution	5
1.7	zDir V2.0 - Files in this package	5
1.8	zDir V2.0 - Requirements	6
1.9	zDir V2.0 - Usage and options	7
1.10	zDir V2.0 - Installation	12
1.11	zDir V2.0 - History	12
1.12	Ni - a revolution in CLI/Shell commands.	13
1.13	zDir V2.0 - About the author	14
1.14	zDir V2.0 - Known problems	14
1.15	zDir V2.0 - Acknowledgements	15
1.16	DIKU - http://www.diku.dk/	16
1.17	xpk - The Amiga compressor system.	16
1.18	PowerPacker - the packer of the past.	16
1.19	MultiUserFileSystem - The Amiga MultiUser security system.	17
1.20	xfdmaster.library - the Amiga unpacking library.	17
1.21	XFH - the xpk-filehandler.	18
1.22	FileID.library - a fileidentifying library.	19

Chapter 1

zDir

1.1 zDir V2.0 (16.8.97) by Finn Nielsen - ©1997 ZzNyYd! Design.

zDir V2.0

The Ultimate Dir-command

Introduction

Why use zDir instead of some other dir listing util?

Background

Why did I make this?

Features

A list of some of zDir's features.

Disclaimer

If something goes wrong don't blame me.

Distribution

zDir is FREeware.

Files

These files should be in the archive.

Requirements

What does zDir require?

Usage/Options

Description of options and stuff.

Installation

How to install the blasted thing.

History

What has happened in the past?

Author

Where to send bugreports.

Problems
When it's not my fault.

Thanks
Acknowledgements.

1.2 zDir V2.0 - Introduction

zDir V2.0 - Introduction

When working a lot with a command line interface a good dir listing util is a must. However the output from the List command, supplied on the workbench disks, fills to much (I don't always need to know the state of the protections bits and creation date and time), the Dir command (also from the wb disks) doesn't give enough information and further more it simply looks ugly and it sucks, and so does most other dir listing utils.

Some dir listing utils, such as xDir, xfdList and MList, comes in a software package and exists only to support that special software package, and, unfortunately, lack a lot of functionality.

So here is zDir, a Dir listing util that has all the functionality that is required by a real Dir command, a nice look, a list option + it supports the software packages

```
xpk
,
xfd
,
fileid
and
MultiUser
.
```

Please take a look at the
Features
that zDir offers.

1.3 zDir V2.0 - Background

zDir V2.0 - Background

A long time ago when my friends and I were still using Kickstart 1.3, I used C='s List command in my CLI and one of my friends used something called "Master Dir", and we were happy with that

The shell always had a special feature, that the CLI did not have: the ability to make commands resident. When working with disks on 1.3 however installing the shell required extra disk space, so we just used the CLI.

Then came Kickstart 2.0. In this the shell was part of the ROM and so required no extra disk space, so now we could take advantage of the resident

command. This was all fine for me because I used List, but my friend who really liked the look of "Master Dir" was disappointed of it's inability to be made resident because it wasn't pure. So he gave me a copy of "Master Dir" and asked me to make a pure dir command, with the same look. I did and I also made a few other improvements and the first version of zDir was born (back then called "Ultimate Dir" or just "Dir").

In the summer of 1995 my harddisk at that time (a 210 MB seagate) crashed and I lost (among other things: real men *don't* make backups :-)) the source for the latest version of zDir (V1.42) and as a result I decided to drop the project. However urged by some of my friends (and also because I was not satisfied with V1.42 anymore) I decided some time ago to do a complete rewrite of zDir, implementing all the functionality missing from the previous versions + support for the software packages

```
xpk
,
xfd
,

fileid
and
MultiUser
.
```

This was initially meant to be programmed in assembler like the previous versions, but due to assembler being a pain I never really got around to doing any work on it. Then in the summer of 1996 I learned C and C++ and during the last 2 weeks of my summervacation I did most of zDir V2.0, but this time in C, so basically zDir has been nearly finished for a year now (only the recursive stuff were missing).

During the following year I had to much else to do (studying Computer Science at the University) so only bugfixes and very small changes were made to zDir. But now (during my summervacation in 1997) I've finally gotten around to doing the last part of zDir, and this is the result.

1.4 zDir V2.0 - Features

zDir V2.0 - Features

zDir can replace:

```
\textdegree{} C='s Dir.
\textdegree{} xDir (from the
    xpk-package
).
\textdegree{} xfdList (from the
    xfd-package
).
\textdegree{} WhatIs (from the
    FIDLIB-package
).
\textdegree{} MList (from the
    MultiUser
    -package)
```

Features:

```

\textdegree{} Says "***
                Ni
                ." when you break it.
\textdegree{} 100% pure, can be made resident.
\textdegree{} Uses efficient memory-pools.
\textdegree{} Very fast assembly written sorting algorithm.
\textdegree{} Shows the full path of the directory that is read.
\textdegree{} Has a much nicer look than other dir-listing utils.
\textdegree{} Can output disksizes in Kb or MB.

```

Features accessible via options

```

:
\textdegree{} List option (recognizes and ignores
                XFH
                -comments).
\textdegree{} Recursive option (can recursively read an entire directory tree).
\textdegree{} Sizecheck dirs option.
\textdegree{} Recognizes
                xpk
                -packed and
                PowerPacked
                data-files internally.
\textdegree{} supports
                MultiUserFileSystem
                (group & other protection bits + owner/group field).
\textdegree{} Supports
                xfdmaster.library
                and
                FileID.library
                .
\textdegree{} Can recursively scan a directory tree for a specific filename
                or filecomment.

```

What zDir doesn't have:

```

\textdegree{} An LFORMAT argument: zDir is a Dir-command not a List-command!
                When you need LFORMAT use List or something.

```

1.5 zDir V2.0 - Disclaimer

zDir V2.0 - Disclaimer

zDir is provided "AS IS" and by using zDir, you accept the FULL responsibility for any damage or loss that might occur through its use or the inability to use it. The author of zDir can NOT be held responsible for any damage or loss of data that might happen through the use or misuse of this software. In particular the author can not be held responsible for any problems that arise due to a bug in any of the external libraries used.

I myself and a friend of mine have been using zDir V2.0 for almost a year now and I have corrected all problems and bugs that we have found in it, so I guess zDir is pretty stable by now. zDir works fine on my system and should work on all V37+ systems, but there is no guarantee that it'll work

on your particular system.

1.6 zDir V2.0 - Distribution

zDir V2.0 - Distribution

zDir is FREEWARE but also Copyright by the author . That means you are allowed to redistribute this package as long as you follow these rules:

1. Any re-distribution has to include *all* files in this archive, without any modifications. The files are listed here , if any of them are missing, you should consider getting your software from another source.
2. zDir may be freely distributed via BBSs, InterNet/UseNet, software libraries such as Fred Fish's and Aminet CD-ROM, and other similar electronic channels. This includes that the author(s) of any other freely distributable software package may include the zDir package with that software.
3. Disk magazines and services that charge extra for file transfers may NOT distribute it without written permission by the author .

1.7 zDir V2.0 - Files in this package

zDir V2.0 - Files in this package

This is the list of files, that I, the author of zDir, have put in this archive:

```
file_id.diz  BBS identification
zDir        The binary itself
zDir.guide  This documentation
zDir.readme The .readme file for Aminet
Authentication.asc  Contains the md5 sums for the above files.
```

Should any of these files be missing from the archive, or any of the md5 sums fail, you should seriously consider getting your software from another source.

So you can check that the md5 sums are valid I've signed Authentication.asc

with my PGP public key below:

```
Type Bits/KeyID      Date           User ID
pub  1024/AB1AB0AD 1995/11/28 Finn Nielsen <zznyyd@diku.dk>
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
```

```
Version: 2.6.3i
```

```
mQCNAjC7XogAAAEALzi4tZpf2DrVicNYgI9hB3nPImYqRQWnaCM9Vin/qnPjM5O
P3plHnZct2fQwv++7YRxxXGyF4zZlXjey5z8tu5Mid0GhtHqCybzaThHLZ9DI0jU
0FYxlEBixbr7zBTrSqMC09lBrqmCGaDXYfaJmHFkLvchL1BwpHM9SeGrGrCtAAUR
tB1GaW5uIE5pZWxzZW4gPHp6bnl5ZEBkaWt1LmRrPokAlQIFEDC7XsVzPUnhqxqw
rQEBmAQEALJNfdV3qZvXE4nsjT4ynf+GHIzFr2lWdsExZzu+9e74lKS0S+pWV0Q0
vnfNqGgQEGnUm5ZwBPF0hhiro5Y4tNSpdY6g0SLqq9R1v0maZLaDrwjY+huL3tVz
ltQcU006v92g4ynGmWqTVRU4FcYBr2TxmaJiKCV8oaBR893dRjym
=Ub/j
```

```
-----END PGP PUBLIC KEY BLOCK-----
```

This key is also available at key servers and at my WWW homepage at <http://www.diku.dk/students/zznyyd/pgp.html>.

1.8 zDir V2.0 - Requirements

zDir V2.0 - Requirements

Normal usage and xDir-mode:

```
\textdegree{} An Amiga with kickstart V37 (2.04) or greater.
```

The Xtra option (-X[n]):

```
\textdegree{}
    xfdmaster.library
    V37 or greater (not included)
\textdegree{} + optionally
    FileID.library
    V7 or greater (not included)
```

The MultiUser option (-m):

```
\textdegree{}
    multiuser.library
    V39 or greater (not included)
```

Note that while zDir refuses to run on pre-V37 systems, it will simply issue a warning (and ignore the offending option) if you have chosen an option that requires xfdmaster.library, FileID.library or multiuser.library without having it in your Libs: drawer. zDir will work fine without these libraries and will only try to load them when told to do so via the

options

.

1.9 zDir V2.0 - Usage and options

zDir V2.0 - Usage and options

Usage:

Usage is simple, just use zDir like most other Dir-commands. Type "zDir ?" to get a small overview of the options. To read several directories just put them after each other and zDir will read them one by one, if you don't specify any directories zDir will read the current directory. If you enter a name with spaces in it, you'll have to enclose that name in single or double quotes or zDir will think that it's more than a single name.

Options can be anywhere on the command line and starts with a '-'. When an '-' is encountered zDir will start parsing options. zDir accepts options both the 'lha-style' like:

```
zDir -loR mydir
```

and the 'lzx-style' like:

```
zDir -l -o -R mydir
```

and mixed like

```
zDir -lo -R mydir
```

and even

```
zDir -l mydir1 -o mydir2 -R
```

Note that zDir parses all options before reading the directory(s) and that the option(s) specified last override the option(s) specified first (except for the -C and -F options). All directories specified in the same line will be read with the same options active. All options are case-sensitive, so while "zDir -m" is ok, "zDir -M" will fail.

To read a directory where the name starts with a '-' just enclose the name in single or double quotes or use the -- option. If you get the 'unknown option' error you probably just forgot to that, for example:

```
zDir -mydir
```

would fail because there exists no -y option, but all of these combinations

```
zDir "-mydir"  
zDir '-mydir'  
zDir -- -mydir  
zDir -- "-mydir"  
zDir -- '-mydir'
```

would work.

Options:

Here is the list of all zDir's command line options. Note that the command line options are all case-sensitive and that all options are toggles with some sideeffects (except the -C and -F options).

-- stop option parsing

This option will stop all option parsing. This option *must* be followed by a space character.

Following this option everything on the rest of the command line is assumed to be file- or directory-names.

-l List-mode

This switch turns on zDir's list-mode. In this mode zDir's output will be very similar to List only with a little nicer look and an extra field for extra information.

The extra information can be one of the following: "(Lnk)" if the entry is not really a file or a directory but a hard- or soft-link to one, "(Dir)" if the entry is a directory, "(XFH)" if entry is a file that has an

XFH

-comment attached to it (XFH-comments will not be shown) and finally "(...)" if the entry is just an ordinary file.

This cancels the -m, -x and -X[n] options.

-L List dates (list-mode)

Same as -l except the date field will always contain the date and never a substitute such as "Yesterday", "Monday", "Future" etc. and -m is not turned off.

This cancels the -x and -X[n] options.

This always turns on the -l option.

-c ignore Comments (list-mode)

Same as -l except zDir will ignore all comments and -m is not turned off.

This cancels the -x and -X[n] options.

This always turns on the -l option.

-m MultiUser list (list-mode)

This option turns on zDir's

MultiUser

-mode. When this is on zDir

will also show the group and other protection bits + the owner/group field. Note that the MultiUser 'u'-protection-bit will be shown in place of the AmigaDOS 'h'-bit. This always turns on the -l option.

If multiuser.library V39+ isn't available this option is ignored (except for turning on the -l option).

This cancels the -x and -X[n] options.

-x xDir-mode

This switch turns on zDir's xDir-mode. In this mode zDir will recognize

xpk
-packed and
powerpacked
datafiles and will show you

both the filesize and the original/uncrunched size of the file + it will calculate the ratio by which the file was crunched and inform you which packer was used.

This cancels the -l, -c, -m, -d and -X[n] options.

-X[n] Xtra

Same as -x except in this mode also

xfdmaster.library
V37+ (and

optionally

FileID.library
V7+) will be used to recognize files.

The output format is a little different because xfd still does not inform you of the uncrunched filesize (except if you really uncrunch the file), this is IMHO almost bad enough to be considered a bug in xfd.

The n is optional and can be either 1, 2 or (the logical OR of 1 and 2) 3, if n is 1 zDir will tell xfdmaster.library to enable it's external libraries to recognize files. If n is 2 then zDir will also use FileID.library (if it's available) to recognize files, and if n is 3 zDir will do both of 1 and 2.

This cancels the -l, -c, -x, -d and -m options, however if xfdmaster.library V37+ isn't available zDir will fall back to using the -x option instead (in this case FileID won't be used either).

-i no disk Info

If this option is specified in the command line zDir will not print out it's usual disk information at the end of a directory listing. This cancels the -s option.

-R Recursive scan

This option tells zDir to read every directory encountered recursively.

This cancels the -f, -d and -S options.

-f only Files

This option tells zDir to only show files. This option cancels the -S, -R and -d options.

-d only Dirs

This option tells zDir to only show directories. This option cancels the -x, -X[n], -f, -s, -S and -R options.

-s show dir Size

This option tells zDir to sum up all the file sizes in a directory. This option cancels the -d, -i, -x and -X[n] options.

-S Sizecheck dirs

This option tells zDir to recursively scan each directory encountered summing up all the file sizes. In this mode all directory names are appended with a '/' to allow you to better tell files apart from directories. Also in this mode the fields Files, Dirs and Total will display the total number of files and dirs encountered. In this mode pattern matching will be disabled.

This option cancels the -f, -d, -x, -X[n] and -R options and unless -i is in effect sets the -s option.

-o sOrting

This switch toggles whether zDir should sort all the names or not. This is mostly useful for turning on sorting in list/xdir-modes. By default sorting is off in list/xdir-modes. Sorting is always on in dir-mode.

-E Escape-codes

This switch toggles whether zDir will output any escape-codes to the console. By default zDir prints files in color 1 (black) and directories in color 3 (blue), but when redirecting output to a file it can be useful to turn escape-codes off.

-F<pattern> : Find <pattern>

With this option you can tell zDir to recursively scan a directory for a particular file or directory which matches the pattern. This cancels all other options except for -m, -E, -c and -i. This option can only be canceled by -C.

-C<pattern> : find <pattern> in Comment

With this option you can tell zDir to recursively scan a directory for a particular file or directory whose comment matches the pattern. This cancels all other options except for -m, -E and

-i. This option can only be canceled by -F.

Settings in the 'zDir' environment variable:

zDir also has a environment variable called (amazingly enough) 'zDir'. By setting this variable you can change some of zDir's default behaviour. Use the 'Set' or 'SetEnv' command to set these. Local variables (Set) override global variables (SetEnv), so you can have different settings in different shellwindows at the same time.

Examples:

```
Set zDir "columns=4 kb=50 mb=50"
SetEnv zDir "kb 150"
```

To save the settings use 'SetEnv' and then "Copy Env:zDir EnvArc:".

If there are any faults in the zDir variable all settings in it will be completely ignored.

Here follows the list of options that are recognized in the environment variable. Environment variable options are case-insensitive (unlike the command line options).

COLUMNS=<n>

This determines how many columns zDir will print out a directory in when in Dir-mode. The valid range is 2 thru 20. The default is 3.

WIDTH=<n>

This determines the width of the individual columns when in Dir-mode. The valid range is 11 thru 41. The default is 22.

KB=<n> and MB=<n>

This affects only the 'Dir size', 'Disk Size', 'Disk Used' and 'Disk Free' fields and determines how many Kb/MB is required before shifting to that notation. For example setting 'kb=50' would mean that there would have to be 50 Kb free on the disk before zDir would express the 'Disk Free' field in Kb's instead of in bytes.

Setting these values to 0 or not setting them at all disables this feature.

OPTS=--<opts>

This allows you to specify some default options for zDir. <opts> are the same options that you can specify on the command line. When using this option zDir will interpret it almost as if the options were the first thing on the command line, the only difference is that the -- option will always give an error and that the options given here *must* be specified together (use '-Eo' instead of

'-E -o').

So if the zDir variable is set to "OPTS=-Eo" the line 'zDir foo' will be interpreted the same as 'zDir -Eo foo' without the OPTS keyword.

If the options are not prefixed with a '-' they are simply ignored. That is 'OPTS=Eo' will be ignored, but 'OPTS=-Eo' will work.

This option is mostly useful if you want sorting automatically turned on in all list-modes and/or you prefer that zDir never prints any escape codes. You can also make zDir ignore comments in list-modes with 'OPTS=-cl'.

WARNING: The same routine is used to parse these options as the command line options, this *will* allow you to set yourself up for getting into trouble. For example specifying -f, -d, -F or -C here is asking for trouble.

In some cases defining an alias is better than using the OPTS keyword, examples of this are the -m, -x and -X options.

1.10 zDir V2.0 - Installation

zDir V2.0 - Installation

To install zDir you only have to do the following:

1. (ACTION) Copy zDir to C:zDir to install zDir.
2. (ACTION) Protect C:zDir +p to set the pure bit.
3. If you use the Shell a lot it is a good idea to make zDir resident during startup (insert "Resident C:zDir Add" in your S:User-Startup).
4. Also you might wish to set some options in zDir's environment

variable

.

It's that simple.

I myself have renamed C='s Dir to WBDir and copied zDir to C:Dir.

1.11 zDir V2.0 - History

zDir V2.0 - History

V1.0 - V1.42 (30.03.95):

Ancient history. V1.42 can be found in Aminet:util/cli/Req.lha, no previous versions have been publicly released (didn't have internet access then). V1.42 is the last version to support pre-V37 Kickstarts.

All the V1.xx versions contain a small bug, causing them to refuse to run when Mungwall was running, this has been fixed in V2.0

V2.0 (16.8.97):

can't handle such behaviour, do not try to break zDir.

1.13 zDir V2.0 - About the author

zDir V2.0 - About the author

My name is Finn Nielsen, I'm a student at the University of Copenhagen where I'm studying Computer Science and Mathematics.

For bug reports, comments and suggestions regarding zDir you can contact me at anyone of these addresses:

```
Internet: zznnyd@diku.dk
FidoNet : 2:235/314.145
AmigaNet: 39:141/105.145
```

Before you report a bug, please make sure that the bug you've found isn't one of the

Known problems

.

1.14 zDir V2.0 - Known problems

zDir V2.0 - Known problems

Here I've listed some problems that can arise with zDir. All of them will usually be caused by something wrong with your system or because you didn't read the

Usage
chapter properly.

If you write to me and it turns out that your problem could have been solved by reading this chapter and the Usage chapter, you can expect a rude reply (I really hate it when people turns their brain off and report bugs that aren't there because they can't read a manuel). If, on the other hand, you report a real bug or comes with some good suggestions or constructive criticism regarding zDir, I'll be very polite.

Here we go:

\textdegree{} zDir always just quits without doing anything and when you use the \leftrightarrow 'why' command is says 'invalid resident library' (Result2 returncode is 122).

Your Kickstart is pre-V37. UPGRADE NOW...

\textdegree{} zDir always fails with 'Error 118: wrong number of arguments', ' ←
 Need arg
 for -F option', 'Unknown option -<something>' or some other message or
 maybe just behaves weird.

Check out the zDir variable, maybe you've specified the OPTS keyword with
 some bad options (or maybe a friend is playing games with you... I'm
 certainly going to see if I can get away with it on one of my friends
 computer).

If the OPTS keyword isn't specified be sure to say so in your letter or
 I'll assume that this is the problem (and you'll just get a rude reply).

 \textdegree{} When setting COLUMNS larger than 3 zDir writes out more than fits ←
 on one
 line.

To solve this problem you have to set WIDTH yourself, zDir doesn't check
 the width of you window but relies on you to make some sensible settings.
 This allows people who use a thinner font than topaz or use a high screen
 resolution to fill out the empty part of their windows when using zDir, I
 use a thin font myself and the settings COLUMNS=4 and WIDTH=22 works
 perfectly for me.

 \textdegree{} zDir crashes when using the -m option:

Do you have a copy of multiuser.library in your LIBS: drawer but haven't
 got the

MultiUserFileSystem
 installed ?

multiuser.library was never meant to deal with the situation where the
 MultiUserFileSystem isn't installed and as such it may or may not crash on
 such systems, meaning that on such systems any program utilizing
 multiuser.library may crash upon opening or using the library.

Solution: If you haven't got the MultiUserFileSystem installed then
 you should remove multiuser.library from your LIBS: drawer.

 \textdegree{} When using the -X option zDir doesn't recognize a crunched file ←
 that
 should be recognized by a
 xfd
 sublibrary.

Have you been using the -X option without specifying a number or with -X2 ?
 zDir only tells xfd to use it's sublibraries with -X1 or -X3.

1.15 zDir V2.0 - Acknowledgements

zDir V2.0 - Acknowledgements

I send my regards to the following people:

Tue B. Petersen For making me do the first versions of zDir and moral support.

Brian Nielsen For moral "support".

Monty Python For saying stuff like "
Ni
" in films.

1.16 DIKU - <http://www.diku.dk/>

DIKU is an abbreviation of "Datalogisk Institut, Københavns Universitet", in english "Department of Computer Science, University of Copenhagen".

You can visit DIKU on the WEB at "<http://www.diku.dk/>".

1.17 xpk - The Amiga compressor system.

xpk - The Amiga compressor system

For a long time, there have been various compression programs for different purposes on the Amiga. Most of these are incompatible with each other and not every application supports every packer. For some time

PowerPacker

by Nico Francois was the most supported packing algorithm, but nowadays PowerPacker is somewhat outdated and it is quite inferior to some newer algorithms.

In comes xpk. xpk is a compressor system that lets an application support any compressor- and/or encryption-algorithm that there exists a xpk-sublib for in a transparent way. By adding xpk-support to an application, a programmer automatically add support for any compressor/encryption algorithm that there exists an xpk-sublib for. And by implementing a new [powerful] algorithm as an xpk-sublib a programmer can instantly gain support for his/her algorithm in a wide range of applications, without having to introduce a new standard.

xpk has introduced a standard for compressor- and encryption-algorithms on the Amiga.

xpk is written by Urban Dominik Mueller, Bryan Ford, Christian von Roques and Dirk Stöcker.

1.18 PowerPacker - the packer of the past.

PowerPacker - the packer of the past

PowerPacker by Nico François was one of the most popular packers, if not THE most popular packer, on the Amiga for a long time (and is infact still very popular among some people). However nowadays it is somewhat outdated and has been superseded by the far superior

```
xpk
-compressor-standard.
```

PowerPacker is infact quite inferior to some of the more powerful algorithms implemented as

```
xpk
-sublibs.
```

I personally consider PowerPacker obsolete.

1.19 MultiUserFileSystem - The Amiga MultiUser security system.

MultiUserFileSystem (Aminet:util/misc)

Copyright © Geert Uytterhoeven, All Rights Reserved.

The following text is from the MultiUser.doc from the MultiUser-package:

You've got an Amiga with Kickstart 2.04 or higher and several people are regularly fooling around with it ... Last week your sister deleted your 20MB JPEG collection by mistake and you don't want this to happen again ... Well, here's the answer: MultiUser!

MultiUser allows you to create a *IX-like environment where several users live together in harmony, unable to delete each others files, unable to read those private love-letters of other users ... And this even if several users are working on the machine at the same time (on a terminal hooked up to the serial port) ...

You are the sole user of your computer? Well, make sure it stays that way by installing MultiUser! People without a valid login ID and password won't be able to access files you have made private with MultiUser. If you make all files private (not readable for others), the only useful thing they could do, is boot from a floppy ...

And ... you do not have to reformat your hard drive!

1.20 xfdmaster.library - the Amiga unpacking library.

xfdmaster.library (Aminet:util/pack)

Copyright © Georg Hörmann.

The following text is from the README file from the xfd-package:

INTRODUCTION

Almost every Amiga user has some crunched files on his/her disks, either

Powerpacked

docs or crunched demos, utilities or whatever. But what to do when you need the files to be uncrunched, and you don't call the cruncher your own (or it doesn't support decrunching)? Enters XFD.

FEATURES

This software package allows you to decrunch almost every packed file known on the Amiga. It consists of the xfdmaster.library as the brain and a couple of programs that offer certain functions to the user.

The xfdmaster.library is a standard Amiga shared, runtime library. It works with Kickstart V33+ and offers applications the possibility to directly support any crunched files.

Other than the predecessor decrunch.library, the xfdmaster.library has a more flexible interface, offers the possibility to decrunch already loaded program segments, supports external decrunchers, is able to unlink so-called 4EB9-linked files and can scan for crunched data in any memory area.

1.21 XFH - the xpk-filehandler.

XFH - the xpk-filehandler

No matter how big a harddisk you buy, at some time it always gets filled, but if you pack your files on harddisk they will use less disk space and then maybe you'll have room for some more.

XFH-handler is a DOS handler which uses

xpkmaster.library

to provide

transparent access to compressed files in a given directory or partition. All compression/decompression is done automatically by the handler when files are read or written. Compression is optional and may be switched at any time, allowing for fine control over storage of data. The compression method may be changed at will. Decompression is always automatic, you don't have to care about which compressor was used to create the files.

XFH is the Amiga equivalent of things like Stacker and DoubleSpace on the PeeCee. Sometimes files packed with

xpk

has a XFH-comment, which XFH uses

for faster accessing of directories (the XFH-comment tells XFH the uncompressed size of the file, so XFH won't have to open the file to find that out).

XFH is written by Kristian Nielsen and Matthias Scheler.

1.22 FileID.library - a fileidentifying library.

FileID.library - a fileidentifying library

FileID.library is a shared library for identifying files by checking it's contents.

Some programs already exist (ID, WhatIs, etc.) doing that job. But other programmers can't use it's services for their own programs. This is the goal of FileID.library.

FileID is written by BLOODROCK of SYNDICATE
