

# **ViewDIZ**

Mikael Lund

Copyright © Copyright 1997, 1998 Mikael Lund.

---

**COLLABORATORS**

	<i>TITLE :</i> ViewDIZ		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Mikael Lund	July 7, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>ViewDIZ</b>	<b>1</b>
1.1	main	1
1.2	intro	2
1.3	descriptions	3
1.4	install	3
1.5	requirements	4
1.6	xfx	4
1.7	rdslib	4
1.8	usage	5
1.9	newfiletype	7
1.10	options	8
1.11	buttons	9
1.12	opt_killdiz	9
1.13	opt_upperdiz	10
1.14	opt_delreadme	10
1.15	opt_maxduplet	10
1.16	opt_list	11
1.17	opt_edit	11
1.18	opt_add	11
1.19	opt_savenote	12
1.20	opt_savediz	13
1.21	opt_clip	13
1.22	opt_save	13
1.23	opt_noreq	13
1.24	opt_pubscreen	14
1.25	opt_font	14
1.26	opt_from	14
1.27	opt_flushlibs	15
1.28	history	15
1.29	author	18
1.30	cataloghelp	19
1.31	legal	20
1.32	registration	20

---

# Chapter 1

## ViewDIZ

### 1.1 main

```
W E L C O M E   T O   V I E W D I Z  
C O P Y R I G H T   1 9 9 7 - 1 9 9 8   B Y   M I K A E L   L U N D
```

Please chose an issue:

Introduction  
What is it ?

File descriptions  
Story about descriptions.

Requirements  
What you need.

Installation  
How to install on your system.

Usage  
How to use it.

History & Future  
IMPORTANT: CHECK OUT THE FUTURE !

Translating info  
How to translate into your language.

Author  
A few words from the Author.

Disclaimer  
Legal stuff.

---

FREE REGISTRATION  
Check it out !

## 1.2 intro

Introduction:

ViewDIZ is a small utility that can show file descriptions from a great variety of files. It is operated by requesters which makes usage very comfortable and a perfect solution for Directory Opus etc.

The file description is obtained in the following priority:

- 1) File\_ID.DIZ
- 2) Aminet .readme file (short-description)
- 3) Filenote
- 4) Version-info (\$VER: / RomTag)

You can chose to edit the information and save it as File\_ID.DIZ or Filenote. You will also have the option to perform a certain action on the file, like unpack a DMS file or play a soundmodule and so on.

ViewDIZ will also work as a powerfull CLI-tool to manipulate file\_id.diz inside archives. It have got loads of arguments that will let you do virtually anything with the description.

List of some features:

- \* Easy-to-use, nice-looking GUI (optional).
- \* Unlimited filetypes (modular design).
- \* Aminet.readme, Filenote & version support.
- \* HTML title support.
- \* File\_ID.DIZ -> Filenote option.
- \* Aminet.Readme -> File\_ID.DIZ option.
  - \* Editing/saving of File\_ID.DIZ.
  - \* Extracting functions on archives and DMS/xFX files.
- \* 100% configurable View/Play/Read/Edit functions.
- \* Direct ARexx support for 5 well-known music players!
- \* Clipboard support.
- \* Locale support.
- \* Description-list creator.
  - \* Binary filetype check on unknown suffixes.
- \* System friendly; Can be made resident.
- \* It's absolutely free!

File processing is done via external modules - below is a list of the included modules:

- \* Archives: Lha/Lzh, Lzx, Zip, Arj, Tar
  - \* Diskpackers: Dms, Xfx, Dsq
  - \* Text: Txt/Nfo, Html
  - \* Pictures: Iff, Jpg, Gif
  - \* Music: Mod/Med, Mus (exotic).
-

## 1.3 descriptions

About file-descriptions:

In these modem-days worldwide transfer of files can be done very easy. What isn't easy though is to keep track of all the files you have downloaded. Most BBS'es only support the old ms-dos filesystem which only allows you to have a filename of 8 chars and a suffix of 3 chars. Obviously this "8.3" filesystem can't be used to create long and meaningful filenames... What does "MDS-QW11.LHA" mean to you ?!

Hmm, well wouldn't it be nice if you could get a longer and more saying description of those meaningless names ?

Yes it would - and you can! What is typically used on BBS'es is something called FILE\_ID.DIZ (Short: "FID")

This expression is just a synonym for appending an ordinary text to the original file. This text mainly describes what the current file contains; in other words, it's a file description (!)

Not all filetypes can contain the file\_id.diz description - it's mainly used on files that are somewhat compressed like LHa, GIF, DMS etc.

The trick is now to view this description, hidden somewhere in the file. Now ViewDIZ comes in! It's main feature is to view such a description - see

usage  
for details.

Sometimes, however, files doesn't contain any such file\_id.diz description and you must look elsewhere:

- \* Files uploaded to the internet-archive called Aminet always comes with an extra #?.Readme file that contains a single lined file description.
- \* All files under AmigaDOS can hold a filenote. This note can contain a 80 chars long file description. It's a very neat way of keeping a description as it can be viewed directly from CLI. Unfortunately this note is lost when uploading to a BBS/Aminet etc.
- \* All files can contain a version-string. This string tells you the name, version and date of creation. Sometimes more informations are given.
- \* Certain files such as the HTML-files used at the WWW contains a title. This will very often tell you what the HTML-document is about.

ViewDIZ searches for all the above description types - and with priority as listed. Now go try it - it's easy!

:^)

## 1.4 install

Installation:

---

- Doubleclick on Install.

Everything  
needed  
will be checked!

## 1.5 requirements

Requirements:

Files stated in bold are included in this package.

- AmigaOS 3.0+
- Active RexxMaster
- Reqtools.library v38+
- REXXreqtools.library
- REXXdosupport.library (  
note!  
)

Files used by external modules:

- Lha, Lzx, Zip, Unzip, DMS, DMSdescript, ARJ\_FId, XFDdecrunch, Copy2Clip  
Textract, MagicDIZ, ModuleDescript,  
xFX  
, xDiz

If you miss some files, try and check out the ViewDIZ homepage:

<http://www.amiga.dk/ViewDIZ>

## 1.6 xfx

Using xFX disk-images:

xFX is another 'DMS-replacement' - but in my opinion it's the far best there is. ViewDIZ can add File\_ID.DIZ to xFX files if they are packed using the xFX pack-option. Currently ViewDIZ uses xDiz to get the file\_id.diz, but a new xFX version is on it's way which have got it's own routine.

If you have the xFX in your command path or the xFX application running in the background, you can also unpack xFX files in the same way as DMS-files.

## 1.7 rdslib



RexxDOSsupport.library:

The author of Rexxdosupport.library doesn't want his production to be included on Aminet CD's. The included version of rexxdosupport.library is not the recent version and is, when included in the ViewDIZ package, allowed to be present on the Aminet CD-ROM's. The full archive can be found at "aminet/util/rexx/rexxdosupport.lha"

Short: V3.5 update information about rexxdosupport.library  
Type: util/rexx  
Uploader: hartmut Goebel <hartmut@oberon.nbg.sub.org>  
Author: hartmut Goebel <hartmut@oberon.nbg.sub.org>

The current version of rexxdosupport.library is 3.5 and can be found in Aminet.

Since there are no freebie CD's without any costs for software authors, rexxdosupport.library may not be distributed on AmiNet CD. This will change as soon as real freebies are given away again.

So check out for the current version on a Aminet ftp-site.

Regards  
+++hartmut

## 1.8 usage

CLI-Usage:

The usage is really quite forward - You can use it directly from CLI:

```
1:>ViewDIZ File/A [Options]
```

options  
are not required to view a description, they are completely optional (wow!).

When started correctly, a window will open with the description (if any) of the chosen file. Also you will see a range of

buttons  
that'll let you  
perform certain actions on the description/file.

Note: Starting ViewDIZ with no arguments will bring up a help screen where you'll get information about options, filetypes etc.

DOpus-Usage:

A more sophisticated method of starting ViewDIZ would be to use Directory Opus - just create a function with the following properties:

---

Function : AmigaDOS ViewDIZ {f}  
Flags : Run asynchronously

If you have Dopus5+ on your system, the ViewDIZ installation script will install some example-buttons for you.

Customizing:

ViewDIZ 2.0+ is build up by filetype-modules. A module is a small AmigaDOS script that handles a specific filetype.

Each time ViewDIZ recognizes a filetype it will run it's module which will then take care of description extraction, saving, stripping and execution (play, read etc.).

In this way unlimited filetypes can be added to ViewDIZ without changing the main program. See here how to

add a new filetype.

Besides the modules, you will find some more files in Env:ViewDIZ/ ↔

- Below

is a short examination of the various configurations files:

Editor.s Script executed when you press "EDIT".

It's important that the used editor do not detach itself from CLI. CygnusED do this if it's not started with the "-Keepio" or "-sticky" option.

Diztabu When using the savenote

option the description will

be saved in the filenote. Before this ViewDIZ will strip the description for some unimportant chars. These chars are specified here.

How to make ViewDIZ faster:

Although I consider ViewDIZ to be pretty fast, you do encounter some waiting - especially on very large archives without File\_ID.DIZ.

Below is a list of things you could do to speed up things a bit:

- Make the following resident: c:ViewDIZ, c:Execute, LHa, LZx
  - LHa, LZx, Textract etc. are to be found in different 680x0 optimized versions. Select one that fits your processor.
  - Place your most frequently used filetype in the top of env:viewdiz/modules.config.
  - Let ViewDIZ be the first Rexx program to be started. Other Rexx programs may add the used libraries in a time consuming priority.
  - Minimize usage of the Maxduplet option.
-

- Buy a 68060 turbocard.

## 1.9 newfiletype

How to add a new filetype:

If you want to add a new filetype to ViewDIZ you must do two things:

1. Create a module.
2. Setup the modules.config

Modules:

Creating a module is very easy - if your skills in AmigaOS script programming is moderate you can take a look at the already made modules.

First of all your module must be able to receive info from ViewDIZ. This is done via AmigaOS argument parsing - the script will simply be called with the following arguments:

```
{file} = Full filename of current file.
{desc} = Full filename of description file.
{cmd}  = read
        write
        delete
        execute
        info
```

The {cmd} argument states what ViewDIZ wants from your module - If {cmd} is equal to "read", your module must try to extract (read) a file description from {file} and then save it as {desc}.

You will also have access to some environmental variables:

"xpkstatus":

1 (true) or 0 (false). If = 1 then the current file is xpk-packed. This is a read-only variable.

"savenote":

Tell ViewDIZ to save the description in the file note. It can be used only when {cmd}=write and it's a write only variable. True = 1.

(Actually it works like if you had used the savenote option).

FUTURENOTE: In coming viewdiz releases "xpkstatus" and "savenote" will be changed to LOCAL variables. This is not possible under ARExx.

Modules.config:

---

ViewDIZ have to know to which module a filetype belongs. It'll also need to know what the module is capable of doing. Can it save a description? Can it perform an action on the file ?,, etc.

This is done in the file "ENV:ViewDIZ/modules.config". The data must come in 4 columns:

1. Full path of module.
2. Function-flags, "RWED" - What can your module do ?  
  
R = Read file description  
W = Write file description  
E = Execute an action on the file (play, view, etc...)  
D = Delete file description
3. Execution button. The "E"-flag under 2. must be set to have any effect.  
- if the execution flag isn't set you must specify "n/a".
4. Pattern that the file should match. (Standard AmigaOS notation)

If data in a column contain spaces, you must enclose the data with quotes.

The lines you specify in the modules.config are processed from top to bottom - when a match is found ViewDIZ executes the module. To improve processing speed you should place your most frequently used filetypes in the top of the file.

## 1.10 options

Options:

Add/K

Load (& save) description

Clip/S

Copies description to clipboard

Delreadme/S

Delete Aminet readme files

Edit/S

Pop up editor before viewing

Flushlibs/S

Remove unused libraries

Font/K

Select font (name.font/size)

From/K

Read above options from a file

---

Killdiz/S  
Remove File\_ID.DIZ from file.

List/K  
Output a "BBS-like" description

Maxduplet/N  
Strip duplicate chars in filenote

NoReq/S  
Disables Requesters

Pubscreen/K  
Select Public screen

Save/K  
Saves description to a file

Savediz/S  
Autosave description to File\_ID.DIZ

Savenote/S  
Saves description to filenote

Upperdiz/S  
Convert description to uppercase

And remember: /A = Required /S = Switch /K = Keyword  
/N = Number

## 1.11 buttons

Buttons:

"Edit" Edit description.  
"Save as FID" Force description to file\_id.diz (if possible).  
"Quit" Quit ViewDIZ.

Besides these buttons you will also encounter other buttons depending on which filetype you are processing, for example:

"Extract" Extract files in archive  
"View" View picture  
"Play" Play music file.  
.  
.  
.  
etc.

## 1.12 opt\_killdiz

---

Killdiz/S:

With this switch on, ViewDIZ will try and remove any file\_id.diz description found in the file.

```
1:> Viewdiz myfile.lha killdiz noreq
```

Will remove (if any) the file\_id.diz stored inside the LHa-archive.

If you use Killdiz/S without the Noreq/S, the description will be extracted first - then it will be deleted.

## 1.13 opt\_upperdiz

UpperDiz/S:

This will let you convert descriptions into upper case. Why? Well, I guess this is a matter of style and taste - use it if you like big letters.

When used the uppercase description isn't automatically saved - You must use either

```
SaveDiz/S
,
Savenote/S
or the edit/save-button.
```

Note: UpperDiz will have no effect when used with the  
Add/K  
-option.

## 1.14 opt\_delreadme

DelReadme/S:

When this switch is on, Aminet readme files will be deleted after they have been saved as either filenote or file\_id.diz. The readme file will only be deleted if you chose the "save" button, savediz- or savenote option.

## 1.15 opt\_maxduplet

MaxDuplet/N:

This option is meant to be used in conjunction with the  
savenote  
option. Descriptions that are to be saved in the filenote will  
always be filtered for the chars listed in ENV:ViewDIZ/Diztabu.

But with the MaxDuplet option you can strip the descriptions even  
further! When used strings of duplicate chars will be deleted -

---

this could for example be a string like "-----".  
This is very usefull when you are processing heavy-duty ANSI-art  
descriptions...!

```
1> ViewDIZ myfile.lha savenote noreq maxduplet 3
```

Will strip the description for duplicate chars with a length of 3  
and above. It will then be saved in the filenote.

(The default value is 0, which means no dupletcheck - Checking is a  
real speed killer:( Encourage me to convert into AmigaE/C !)

## 1.16 opt\_list

List/K:

This option will create a BBS-like description. The format is  
Name Size Description, alligned in a neat way. If the specified  
file already exists, the current description will be added.

```
1:> ViewDIZ file1.lha noreq list ram:filelist
1:> ViewDIZ file2.dms noreq list ram:filelist
1:> ViewDIZ mod.tune noreq list ram:filelist
```

Or from DirectoryOpus:

```
Function : AmigaDOS ViewDIZ {f} list ram:filelist noreq
Flags : Run asynchronously
Do all files
```

Got the idea ? That's what I thought! But remember that there are several  
tools that can do this better, for example FiDED by Lacerdance Productions.  
So if you are practicing this a lot I recommend you get such program.  
Let's not forget that ViewDIZ is just a simple description viewer :^)

## 1.17 opt\_edit

Edit/S:

This option will popup the editor/string-requester before the description  
is viewed. It is especially usefull in combination with the "NoReq" option:

```
1:> ViewDIZ myfile.lha edit noreq
```

Will automatically start you editor with the found description. When the  
editor is closed, the description will be saved without any user activity.

## 1.18 opt\_add

---

Add/K:

This option will let you load a description from a file and then automatically save it to the processed file.

```
1:> ViewDIZ myfile.lha Add ram:description Noreq
```

- Will first load the file "ram:description" and then add it, as File\_ID.DIZ, to "myfile.lha". No user activity required.

```
1:> ViewDIZ myfile.lha Add ram:description Savenote Noreq
```

- Will save "ram:description" in the filenote of "myfile.lha". Again no user activity is required.

If you want to add an ASCII header to a serie of files (keeping the old description) you could create a DirectoryOpus button like this:

```
Function : AmigaDOS   ViewDiz {fu} noreq save t:desc
           AmigaDOS   C:Type t:headerfile.txt t:desc to t:newdesc
           AmigaDOS   ViewDiz {f} noreq add t:newdesc
```

```
Flags : Do all files
        Run asynchronously
```

## 1.19 opt\_savenote

Savenote/S:

The description will be filtered and saved in the filenote, even if the description could be saved as File\_ID.DIZ. If invoked without the "NoReq" option you will be able to edit the description before saving (by pressing the 'EDIT' button).

As mentioned the description will be "filtered". This means that certain chars will be removed from the description before saved; fx. ANSI graphics. "ENV:ViewDIZ/DizTabu" contains a list of chars that will be filtered when using the savenote option.

Myfile.lha is thought to contain a File\_ID.DIZ description:

```
1:> viewdiz myfile.lha savenote noreq
```

This will filter and save the file\_id.diz to the filenote without any user activity.

NOTE:

```
Savediz
and Savenote shouldn't be used together
```

- see also

```
MaxDuplet/N
```



## 1.20 opt\_savediz

SaveDIZ/S:

If possible the obtained description will automatically be saved in File\_ID.DIZ - This option is overruled if the description was obtained from the File\_ID.DIZ in the first place.

This feature is very usefull if you are processing Aminet-files:

```
lha2lzx.lha
lha2lzx.readme
```

```
1:> viewdiz lha2lzx.lha savediz noreq
```

Will save the .readme's SHORT-description as File\_ID.DIZ. If you use this feature without the "NoReq", ViewDIZ will automatically save the description as File\_ID.DIZ before viewing the description.

NOTE: Savediz and  
Savenote  
shouldn't be used together.

## 1.21 opt\_clip

Clip/S:

With this option enabled the description will be copied to the clipboards unit 0. You may then use "ALT V" to insert the description in various programs.

NOTE: The description will be copied before any resuesters are popped up - if you edit a description the new one will not be copied to clipboards.

## 1.22 opt\_save

Save/K:

Lets you save the description to a file. Example:

```
1:> viewdiz myfile save ram:my_description
```

Will show the description as normal, but also save it to ram:my\_description.

## 1.23 opt\_noreq

NoREQ/S:

This option means "No Requesters" and will let you perform actions on a

---

file without any user-activity.

```
1:> viewdiz myfile noreq
```

Will do absolutely nothing!

```
1:> viewdiz myfile clip noreq
```

Will copy the description to clipboards without any user-activity.

## 1.24 opt\_pubscreen

Pubscreen/K:

Lets you chose which screen ViewDIZ should open on. Here you must remember that public screen names are case sensitive.

```
1:> viewdiz myfile pubscreen MY_Pubscreen
```

If "MY\_Pubscreen" isn't opened ViewDIZ will open on the screen from which it was launched.

## 1.25 opt\_font

Font/K:

If you dislike the supplied "Jetterm" font you may use another. It is advised though that you use a non-proportional font.

```
1:> viewdiz myfile font topaz.font/11
```

Will use the topaz font, size 11

## 1.26 opt\_from

From/K:

This option lets you read ViewDIZ' options from a file. The options must be stated in one line exactly as when parsed as arguments. Usefull if you are using ViewDIZ with a lot of options and don't want to type them all each time you start ViewDIZ.

NOTE: File/A and From/K can't be used inside the from-file.

Example:

```
1:> echo >t:my_args "font p0t-noodle.font/8 savediz noreq"
1:> viewdiz myfile from t:my_args
```

This will do the same as:

---

```
1:> viewdiz myfile font p0t-noodle.font/8 savediz noreq
```

## 1.27 opt\_flushlibs

Flushlibs/S:

This option is, hopefully, not to be used very often. When used all Rexx libraries that aren't used by ViewDIZ will be FLUSHED - that is made unavailable to the Rexx-server.

Why?? When you call a function from Rexx, you can't specify in which library this function are to be found; The function are searched for in the current library-list, and the first found function will be executed. When having different libraries in the list, it sometimes happens that functions share names - the one with highest priority will be executed. Executing a wrong function will create an error for sure!

If you encounter any problems with ViewDIZ, this switch will might solve it. But be CAREFULL: Flushing libraries will also make the libraries unavailable to other Rexx tasks which might need them...

If anyone know how to handle this problem in a more secure matter I'd be happy if he/she would contact  
me  
.

## 1.28 history

History:

2.1 xx-xxx-xx (status at v2.0 release date)

- Reqtools requesters.
- Filetype modules are fully supported.
- Configuration file works.
- Binary filetype check.
- Primitive wordwrapper for filenotes and aminet descriptions.
- Localvars "xpkstatus" & "savenote" - global vars are used in v2.0 version.

2.0 18-Mar-98

- Major change: Filetype system completely rewritten - Modular design ensures unlimited number of filetypes (User defined).
  - New options: Delreadme/S, Upperdiz/S, Killdiz/S
  - Added hungarian catalog.
  - Cleans up the T: drawer after Lha-readme processing (Better late than never...Thanks Fabio!)
  - Safer Lha/Lzx handling.
  - Autodetect Eagleplayer2/Delitracker2 working directory.
  - Registrationform generator included.
  - Homepage!
  - Documentation layout change.
-

## 1.9 17-Oct-97

- Locale support.
- NEW options: Add/K, Edit/S, Flushlibs/S, List/K, MaxDuplet/N
- File\_id.diz on ARJ, XPK, xFX (packed), MOD/MED (unpacked).
- xFX unpacking using either CLI-command or ARexx-port.
- More error trapping.
- Improved CLI-template.
- Installer-script: DOpus5+ view/read/play functions now supported, more checks added + some minor improvements.
- Viewdiz.small image added for DOpus5 users.
- Removed "Winshots" + documentation for external commands. (Took up too much disk space)
- ENV:Viewdiz/DMSdest renamed to "DevDest".
- HELP-button errormsg fixed.
- Binary Lha/HTML check fixed.

## 1.8 06-Sep-97

- DiskSqueeze support (Read/Write of File\_ID.DIZ).
- NEW options: Pubscreen/K, Font/K, From/K, SaveDIZ/S.
- DMS drives selected via configfile.
- Music prefixes selected via configfile.
- Hippoplayer, AModPlay & Multiplayer ARexx support added.
- Description now scrolled in Eagleplayer window.
- "Forbidden chars" used by "Savenote" selected via configfile.
- "NoREQ" option now works with "Savenote" too.
- Unpack-button for XPK- and Powerpacked datafiles.
- Directory Opus buttons included.
- Install script rewritten (Using the standard Installer) now containing full configuration part ...brrr a tough and dirty job!
- Configuration system completely rewritten - User defined programs now started via scripts -> 100% configurable!
- Now uses rexxdossupport.library to improve functionality.
- Argument handling completely rewritten - Works much more prof. - Save option now supports spaces in destination file.
- Improved documentation (I hope :^)
- Newer version of MagicDIZ included, v2.05 --> v2.06
- Improved errorhandling.
- Binary MED-recognition added.
- Binary THX-recognition didn't work --> Fixed!
- Wrong .Readme code stole precious CPU/DISK-time! --> Fixed! Also Readme's are now searched in arc-subdirectories too.
- Lots of code cleanups and optimizations, again! :^)

## 1.7 27-Jul-97

- SAVENOTE-option added - Saves file\_id.diz in filecomment.
  - CLIP-option added - Copies file\_id to clipboards.
  - SAVE-option added - Copies file\_id to a file.
  - NOREQ-option added - a sort of 'quiet' mode.
  - Custom Texteditor via configfile.
  - External TXT-extractor now used - Welcome to TEXTract ! Textfiles larger than 65k now supported.
  - File\_ID.diz support for guide-files removed due to TEXTract.
  - Binary filetype-check on textfiles to avoid adding File\_ID.DIZ to packed files.
  - Filename no longer need to contain full path.
-

- New design of the help-window.
- Max. linelength changed from 40 to 44 (file\_id.diz "standart").
- Bugfix in picture recognition -> Major speed increasement!
- Small improvements made on several routines.

#### 1.6 08-Jul-97

- File\_ID.DIZ edit/saving for GIF, IFF, JPG, Textfiles.
- Filenote edit/saving for unknown filetypes.
- View/Play option for pics/music/text - Including an easy and fast configfile for customizing the viewer/player.
- Direct ARexx-support for Delitracker/Eagleplayer.
- Aminet .readme file now searched in LHa archive too.
- Description splitting if longer than 40 chars (word orientated).
- Installation script included.
- Documentation now in AmigaGuide format.
- Minor bug removed in aminet.readme processing.
- Bug removed in TXT/HTML processing - These are now being searched. for .readme/filenote & version info too.
- Optimized the code for speed.

#### 1.5 01-Jul-97

- Aminet .readme support.
- HTML Title-support.
- DMS Editing and Extracting.
- Full ZIP support.
- Version search added.
- Binary filetype-check on unknown suffixes.
- DMS processing made faster (Now using DMSdescript).
- Optimized disk routines for speed.
- Fixed a bug when processing LZx files.
- Source no longer included because of unauthorized clones.

#### 1.4 05-Jun-97

- Added my own file\_id.diz extracting routine; this results in some slower DMS-processing, but spaces in filenames are now allowed. GIF, ARJ & ZIP files can't be extracted now (Who cares anyway??)
- Uses filenote-info if no file\_id.diz is found
- Fixed a bug when editing file\_id.diz in LZx-files

#### 1.31 20-May-97

- First public release.

#### FUTURE:

- I am currently working on an AmigaE version. This means that future ViewDIZ versions will be considerably faster, smaller and won't be needing the currently used REXX#? libraries. As I have no previous experience with AmigaE programming, it will might take a while until the next viewdiz version arrives.

The current filetype system build up with external modules will not be changed.

- What would you like??

#### KNOW BUGS/RESTRICTIONS:

Not much should be found here (!). I have found some rather less important

restrictions though:

- Can't write a blank line to filenote. This is because of a missing flag in the rexxreqtools.library - It's not a bug but the flag is simply not implemented.  
Will not be changed until a new version of rexxreqtools.library is released :(
- You might encounter some problems if you have got ARexx applications using the moos.library - Some functions are named in the same way as in rexxdossupport.library... This could make some confusions, depending on priority. I havn't noticed anything on my system though. If you encounter any problems try out the  
FLUSHLIBS/S  
switch.

## 1.29 author

A few words from the Author:

You are welcome to contact me for suggestions, bugs or whatever. If you want a new feature let me know - I'm always open to new suggestions.

As the mentioned e-mail address is a more or less temporary address it might not be available for years to come. If you can't connect you can always mail me via ordinary mail. Everyone will get a reply.

adr: Mikael Lund  
Ægirsgade 66, 2 th.  
2200 København N  
Denmark

e-mail: ki9656@unidhp.uni-c.dk  
lund@kiku.dk (< 200Kb Only!)

phone: +45 35856154 (Scandinavian or english :=)

Translators:

- Svenska, Magnus "MadOne" Andersson
  - Polski, Don Adan / Wanted Team
  - Italiano, Fabio "Dr. G" Vitale
  - Nederlands, Niels Van Willigenburg
  - Deutsch, Alex "Neurodancer" Kunz
  - Magyar, Tamás Tóth
  - English, Mikael Lund
  - Dansk, Mikael Lund
- More catalogs are most welcome!

Greetings: (No particular order)

- Fabio "Dr. G" Vitale
  - Niels Van Willigenburg
  - Paul Morabito
-

- Alex "Neurodancer" Kunz
- Alessandro Pedretti
- Stuart "Kyzer" Caie
- Don Adan/Wanted Team
- Magnus "MadOne" Andersson
- Nicolas Pomarede
- Eagleeye/Defect
- Hartmut Goebel
- Duncan "Dunx" Gow
- Martin Jeppesen
- Roland de Herder
- Nick Christie
- Dreamer/Abyss/100%
- Dexter/Abyss
- Anxious/Rebels

Big thanks to all who have given me response regarding this project.

### 1.30 cataloghelp

#### Translation Instructions:

You are most welcome to translate ViewDIZ into your native language. This is not a very hard nor time consuming job - About 15-30 minutes is all it takes.

If you decide to create a catalog for your language, you should follow the guidelines listed below:

- Get "CatEdit" by Rafael D'Halleweyn  
"Aminet/dev/misc/CatEdit1\_2b.lha"
- Load the supplied catalog description, "ViewDIZ.cd"
- Translate the strings. Try and be as precise as possible - don't use large explanations. Sometimes you will meet the char "\_". This will indicate that the next char will be a keyboard shortcut.
- Use the menu item "Change language" to set your language. (for example "italiano", "deutsch" ... )
- Save the catalog to:  
"Locale:Catalogs/Yourlanguage/ViewDIZ.catalog"
- Try start ViewDIZ as usual - Your language should now be used. Each time you have edited a catalog-file you will have to reset your machine before the changes takes effect in ViewDIZ.
- When everything looks alright, mail the "ViewDIZ.catalog" file to  
me  
.
- Thanks for your support !

As an alternative, you can use "ReCatIt" instead of "CatEdit". It's a bit more difficult to use, but have got more features.

### 1.31 legal

```

*****
*
*                                     *
*                      DISCLAIMER      *
*                                     *
* THIS SOFTWARE/INFORMATION IS PROVIDED "AS IS". *
*                                     *
* NO REPRESENTATIONS OR WARRANTIES ARE MADE WITH RESPECT TO THE *
* ACCURACY, RELIABILITY, PERFORMANCE, CURRENTNESS, OR OPERATION *
* OF THIS SOFTWARE/INFORMATION, AND ALL USE IS AT YOUR OWN RISK. *
* THE AUTHORS ASSUME NO RESPONSIBILITY OR LIABILITY WHATSOEVER *
* WITH RESPECT TO YOUR USE OF THIS SOFTWARE/INFORMATION. *
*                                     *
*                                     *
*****

```

#### Copyright:

- ViewDIZ is Copyrighted © 1997-1998 by Mikael Lund
- rexxreqtools.library is Copyrighted © 1992-1994 Rafael D'Halleweyn.
- rexxdossupport.library is Copyrighted © 1994-1997 by hartmut Goebel
- JetTerm font is Copyrighted © by Neurodancer/100%

#### Distribution:

The ViewDIZ package must be spread via any PublicDomain/Shareware media.

If the ViewDIZ program is included in a commercial product, a free copy of this product must be given to the author  
- see below.

#### Note:

Some of the external programs and libraries have special copyright and distribution conditions - refer to their documentation for details.

### 1.32 registration

#### Free Registration:

ViewDIZ is freeware which means that there are no restrictions and you may use it absolutely free of charge. I would, however, very much like to hear from you if you use it. Tell me what you think about it, new suggestions or whatever. It would mean a



great deal to me and the future of ViewDIZ ...

To become a registered user of ViewDIZ (which is free!) you will have to fill out the formula below and mail it to me. You will then receive an e-mail when new ViewDIZ versions are available.

You can also use the supplied script to generate the orderform - I guess that's the most comfortable solution. (Especially for YAM users - check it out:))

Thanks in advance,

Mikael Lund

--- cut here -----

\$VER: ViewDIZ\_Registration 2.0

NAME:

COUNTRY:

TOWN:

E-MAIL:

CPU:

OS:

COMMENT:

--- cut here -----

---