DiskMaster

Rudolph Riedel

COLLABORATORS			
	TITLE : DiskMaster		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Rudolph Riedel	August 14, 2022	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

DiskMaster

1.1 DiskMaster-Guide Main

```
                ---------------------------------------------------------------------------

                                DiskMaster II
-------------------------------------------------------------------------------
        Copyright © 1997 by
                Rudolph Riedel
                 (rriedel@amiga.on-luebeck.de)
```

DM2 Mailinglists

1.2 The stupid that hacked all this text into his machine...

```
-------------------------------------------------------------------------------
                                    Author
-------------------------------------------------------------------------------
```

Yep, my name is Rudolph, exactly like the famous Reindeer... :^)

I was born 24.01.1972 in Ratzeburg, Germany

My Computer-Career started eons ago with a C64.
This little amazing machine ate a lot of my time as it demanded
more and more from me, resulting in a few demos and a game.
My last handle/group was (is!) Shadowolf/Paralyze.

Then I came across Amigas.

My first system was an used A500 which I pushed to
1meg Chip/ 4meg Fast / 210 HD / A570 CD-ROM / 2 drives and Kick 2.05

My A1200 is now equiped as follows:

2 MB CHIP
16 MB FAST
Blizzard 1230 IV - 50 Mhz 030, no 68882
Quantum Maverick 540
NEC Multispin 4*4 ATAPI CD Changer
Creatix V34+ Modem
Kickstart/WB 3.1

Today I‘m working for a small company that developes and produces
PA systems for showmen.
I finished my aprenticeship in mid January to become a:
"Kommunikationselektroniker/Informationstechnik"
Yep, eat this piece of german buraucrazy and don‘t try to translate it!

If you have any questions to this Guide or to DiskMaster itself,
feel free to contact me at:

rriedel@amiga.on-luebeck.de

Please use "diskmaster" somewhere in the subject, this will help
me to bring back some structure into my pm.

I also like to receive your Startup.DM and ARexx-Scripts for DiskMaster.

And please excuse my sometimes bad style, English is not my native language.

For those of you, that read thru this point,
also notice that I‘m member of the INCAL! party posse!
You may have met me at cool parties like THE PARTY, or BlackBox Symposium.

Dedicated to Angela, the most wonderfull reason for my
problem with getting enough spare time...

1.3 Now we are a team...

```
-------------------------------------------------------------------------------
                                   Co-Author
-------------------------------------------------------------------------------

Name  : Richard Backhaus
age : 20
aka : dragon
groups  : incal and various others

computer
history : - father bought an Amiga 500 in 1987
    - some programs in basic
    - various releases of assembler tools, doors and intros (incomplete):
    - mcdiz
    - dmsinterface
    - dgnstatus (/X)
    - dgncomment  (/X)
    - dgnlogoff (/X)
    - dgnpage (/X)
    - dgndizadd (/X)  *
    - dgndizstrip (/X)  *
    - dgnlogoff (DD)
    - dgnlhachecker
    - dgnlzxchecker
    - txtdiz
    - sidconvert    +
    - perwarp   *
    - dgncrypt    *
    - lhagui    +
    - c64charconverter  +
    - dragondiskmasher  +

    - various releases of amiga-e made tools and doors (incomplete):
      - arclist
      - capsbranch
      - dgndir
      - dgnfront  (/X)
      - gbfcomment
      - dgnstatus (DD)
      - dgnfront  (DD)
    - dgnpwfail (DD)
    - zipdiz

    * = internal
    + = so far unreleased, maybe someday
    - graphics for many groups' scene productions
    - some web pages e.g. http://www.on-luebeck.de/~rriedel
current
system  : - Amiga 1200/Kickstart 3.0
```

2mb Chip/8mb Fast
an old Seagate 540mb harddrive
Mitsumi double speed cd-rom drive
M-Tec mc68030, 28mhz, 68881fpu
USR Sportster 28000
HP DeskJet 510
- soon to be accompanied by a pc/mac

reallife: - doing military service
hobbies : - graffiti, graphics, music, travelling, computers, literature,
saturday nights, politics, philosophy, editor of maybe the
best scenepapermag brainfART

contact : - e-mail: rbackhaus@hotmail.com

For pure Diskmaster reasons please e-mail to Rudolph,
as it is his project and anyway if your mail relates to parts of
my work, he will surely forward this mail to me.

"Do you want to be dope, or do you want to deal it?"

Dedicated to all my true friends

1.4 What`s that all about?

What the heck is DiskMaster and why exists this Guide?

DiskMaster is simply one of the very best directory-utilities ever!

DM is small fast and compatible, easy to use, easy to configure.
And among the best things it comes for free now!

You simply have to try out DiskMaster to become one of it‘s fans!

I came across DiskMaster early 1993, as far as I can remember.
To be honest, someone gave me a copy of this program that
I tried out on my A500 with 1MB and Kick 1.3 .
Half a year later I really was into it and >honest< from then
I tried to buy the program but there was nobody selling it.
We even tried to phone Progressive, this was back in 1994 or so.

In late 1994 I got hold of a copy of DiskMaster 2.1c,
again from an unknown source - no way to track it down.

In late 1995 I bought my modem - That finally opened the door.
As soon I had a connect to the german z-netz, I started writing adverts.
But again no-one was willing or able to sell an original.

It was in October 1996 when I finally made it to the internet.
On November 13. I put my first advert in COMP.SYS.AMIGA.MARKETPLACE .
And the 16.11.1996 became the day, I was waiting for so long.

Jerry W. Rubemeyer from Iowa offered me his DiskMaster 2.0
together with the manual I ever needed so badly (100 pages).
Thanks again, Jerry!

On December 10. I put my first offer to swap Startup.DMs on the net.

The reactions on this and the following adds convinced me that I am
on the right way still supporting this splendid dir-util.
And as it looks like a lot of users received DiskMaster from the Cover-Disk
or from dubios sources like myself, I decided to write this guide to help all
these fellowers out there that are not so happy
to get hold of the original manual.

Additionally the manual is rather brief on some issues and I received a lot
of examples that I can provide to demonstrate the power of DiskMaster.

To my shame I basically copied the manual so far adding only few comments.
But I also provided some not mentioned functions.
The reason for this is that I wanted to have a solid basis first
to which I may attach new ideas easier.

And now that I received the source for DiskMaster, this is no longer a
stand-alone product but the official documentation for DiskMaster II.

I spent and will spend a lot of work on this, so apreciate it!

1.5 Copyright

```
                ---------------------------------------------------------------------------

                                  Copyright
-------------------------------------------------------------------------------
```

DiskMaster II is written by Greg Cunningham

It‘s his baby and I‘m only a kind of step-father!

On 97-03-10 I finally received the source-code from Greg.

So at this point I took over development to bring
DiskMaster to a more stable and even better point.

```
DiskMaster II is now © 1997 by
                Rudolph Riedel
                The current status of DiskMaster is Freeware!
```

And this means that you are allowed to copy and use it for free,
NOT that DiskMaster is Public Domain!

It is not allowed to gain any profit by selling any part of this
distribution in any form.
PD collectors have to ask for my permission first!
And this also includes magazines that want to release DM
on one of their cover-disks!

It is also not allowed to distribute only parts of this package,
all files mentioned in
Distribution
have to be included.

And finally it is not allowed to use file compression methods like crunching
the exec using PowerPacker/StoneCracker/<other Crap> on any file
in the distribution in order to re-distribute them in that form.

A distribution on Aminet CD's is always allowed.
As well as I gave my permission for distribution on comp.sys.amiga.binary.

This Guide is © 1997 by
Rudolph Riedel
(rriedel@amiga.on-luebeck.de)

The current status of the guide is Mailware!

So if you use it, all I request is that you send me an e-mail or
a postcard to give me a little feedback for my work.

My Address is:

Rudolph Riedel
Grüner Weg 63
23909 Ratzeburg
Germany

But please READ this guide before answering and PLEASE be patient,
I read all your mail but I'm very slow in answering now...

1.6 Requester magic...

ReqTools

This aplication is using the ReqTools.library.

ReqTools is Copyright © Nico François and Magnus Holmgren!

1.7 This is really necessarry...

Disclaimer

The author is in no way liable for any damages or loss of data directly or indirectly caused by the information given in this document! Some of the information could be false other could be easily misused.

In other words:
When you try any of the supplied examples on your configuration you do this completly on your own risk!

Like it‘s not my fault when you delete your files, you pushed the button!

DiskMaster itself is provided as is, you use it entirely on your own risk!

1.8 Really no joke...

```
-------------------------------------------------------------------------------
                         How to install DiskMaster II
-------------------------------------------------------------------------------
```

This is one of the simpliest things on earth, so I won‘t spend any time on writing a special installer script.

1. Copy DM2 and DM2.info to your favourite "Tools" directory using the Workbench or DiskMaster itself.
2. Copy DM2.guide to any directory of your choice.
3. If you like, copy the ARexx scripts from DiskMaster/rexx to Sys:Rexx and the tools from ../c to Sys:C

That‘s it!

Additional steps:

1. Start DiskMaster and select "Save Config" from the "Project" menu.
2. *READ* this guide and use a text-editor to customize your Startup.DM. The file "MyStartup.DM" may give you additional ideas.
3. Use an Icon-Ed or even better SwazInfo to select your favourite icon.

1.9 Functions

```
                ---------------------------------------------------------------------------

               DiskMaster Functions -
                  General Information
                ---------------------------------------------------------------------------


                     About

                    Deselect

                     Print

                    SetList
```

AddAutoCommand

Dirlist

PrintDir

SetPattern

AddCommand

Expand

Priority

SetPrinter

AddKeyCommand

Extern

Protect

Single

AddMenuCommand

Find

Quit

Sort

Archive

Font

Read

Status

Auto

HostScreen

Recursion

StdIO

BarFormat

Lock

Rename

Swap

`Batch`

`Makedir`

`Reset`

`TitleFormat`

`Button`

`Move`

`ReqPattern`

`UnLock`

`Check`

`Msg`

`REXX`

`UnMark`

`ChangeCommand`

`NewDir`

`Root`

`Unpack`

`CloseWindow`

`NewScreen`

`SaveConfig`

`Wait`

`Color`

`OpenScreen`

`ScrBack`

`Window`

`Comment`

`OpenWindow`

`ScrFront`

`Confirm`

`Pack`

Select

Copy

Parent

SetDate

Delete

Pens

SetFormat

Removed Functions

1.10 Some notes on the functions described

```
                ----------------------------------------------------------------------------
                              General Information
-------------------------------------------------------------------------------

- All DiskMaster functions can be issued through ARexx.
  (Arexx only functions are marked)
 This is why there is so less documentation in here for ARexx...

- The entire command, including menu, title,
                command
                 string,
  etc., can be no longer than 255 characters.

- The maximum amount of arguments for a function is 29.

- Due to restrictions in the function parser, all function names
  could be savely abreviated down to five characters.
  E.g. "Extern" could be abreviated as "Exter".

- A special link to the
                flags
                 being used.

In the functions below, arguments enclosed in "< >" are required,
while those enclosed in "[ ]" are optional.

My very own comments are marked with "*".
```

1.11 About

```
------------------------------------------------------------------------------
                                     About
------------------------------------------------------------------------------

Function: Brings up the "About"-Requester.

Syntax: About

Arguments: None
```

1.12 AddAutoCmd

```
                ------------------------------------------------------------------------------

                                AddAutoCommand
-------------------------------------------------------------------------------

Function: Adds an AutoCommand to the list.

          When a file double-clicked, DiskMaster attempts to determine
          what type of file it is by matching the first few bytes of
          the file to a list of known patterns.
          If those few bytes match any of the patterns, the corresponding
          AutoCommand will be executed upon that file.

Syntax: AddAutoCmd [data,
                pattern
                ,
                command
                ]

Arguments:

(no arg): If this function is issued with no arguments, a requester will appear
          asking for the proper arguments.

data: The data compared against the first few bytes of the file,
      to determine wheter or not this AutoCommand is to be executed.
      The "?" wild card character may be used to represent
      a single unknown character.
      Control characters are preceded by the "^" character.

      Two special flags may be used in place of the data: TEXT and DEFAULT.
      The TEXT AutoCommand will be executed if no other patterns
      are matched and the file is a text file.
      The DEFAULT AutoCommand will be executed
      if no other patterns are matched.

Examples:


 AddAutoCmd ?PNG,Extern Tools:ppshow
                %s
                After double-clicking on a file, DiskMaster will read the first ↩
                   few bytes
of the file and attempt to match it with the pattern "?PNG",
which is the pattern for PNG pictures.
If the pattern is matched, the command string
"Extern Tools:ppshow
                %s
                " will be executed.

 AddAutoCmd ??????JFIF,Extern Tools:Fastview %s gray

This is the same for JPEG files, only with more unknown characters.

 AddAutoCmd GIF8,*.GIF,Extern Tools:Fastview
                %s
                The file must end with ".GIF" and must match the data pattern also ↩
                   .

 AddAutoCmd ,*.PCX,Extern Tools:Fastview
                %s
                The data doesn't matter, but the filename must end with .PCX to ↩
                   match.
```

1.13 AddCmd

```
                ----------------------------------------------------------------------------

                                  AddCommand
--------------------------------------------------------------------------------

Function: Adds a command to the current command window.

Syntax: AddCmd [title,##,
                command
                ]

Arguments:

title: The text that will appear in the command window to
       identify this command.

##: A two digit number, specifying the foreground and background
    colors to be used to display the Title text.
    The two numbers used are DM color palette register values.

Example:

 AddCmd   DH0:,10,NewDir DH0:*

Adds a command called "  DH0:" to the current command window,
which when activated, changes the current window directory to dh0: .
The text "  DH0:" will be displayed using color 1 as its forground
and color 0 as its background.
```

1.14 AddKeyCmd

```
                ---------------------------------------------------------------------------

                                  AddKeyCommand
------------------------------------------------------------------------------

Function: Add a keyboard command.

Syntax:   AddKeyCmd [key,
                command
                ]

Arguments:

key: The key to be pressed for the command.

Examples:

 AddKey c,Copy
                %s

                %d
                Press "c" to begin a file copy.

 AddKey C,Copy
                %s
                Press SHIFT+C to start a copy that requests dest path.
```

1.15 AddMenuCmd

```
                ---------------------------------------------------------------------------

                                 AddMenuCommand
------------------------------------------------------------------------------

Function: Adds a new menu entry to the current menu strip.

Syntax: AddMenuCmd [menu_name,title,key,
                command
                ]

Arguments:

menu_name: The name of an existing or new menu to which
           the new menu item will be added.

title: The text that will appear in the menu to identify this command.
```

```
key: A single printable character to be used in conjunction with the
     [Right Amiga] key to activate this command from the keyboard.
     The character used is not case sensitive.
     However, special characters such as "&" or "©" will require
     the use of the [Shift] or [Alt] keys to invoke the command.
```

Example:

```
 AddMenu Config,About,A,About
```

Adds an item called "About" under the "Config" menu, which can be activated by pressing [Right Amiga] + [A].

Hint: In the Startup.DM, adding menus prior to opening any windows
               will increase the speed of the script execution.

1.16 Archive

```
------------------------------------------------------------------------------
                                    Archive
------------------------------------------------------------------------------
```

Function: Provides a multi-select feature taylor-made for file archivers.

Syntax: Archive [ArcCommand]

Arguments:

ArcCommand: The external archive command with it‘s normal "Add" arguments.

Example:

```
 Archive "LHA <* -r a"
```

Opens a requester asking for the archive name, then adds all selected files to the existing archive, or creates a new archive file.

* Here‘s a trick I received from Jody Tierney (jtierney@cyberlink-inc.com)
  to "misuse" archive for multi-selection:

```
>- Archive _can_ be tricked into not opening a requester.  The following
> command sends the full path & name of all selected files in the source window
> to "T:Selectedfiles".  Archive "C:Echo >T:SelectedFiles" %s
```

* The only drawback of this is that the first selected File will appear twice.

1.17 Auto

```
                ------------------------------------------------------------------------------

                                  Auto
```

```
-------------------------------------------------------------------------------

Function: Uses auto commands on all selected files.

Syntax: Auto [
                file
                ]

Example:

 AddCmd Auto,20,Auto
                %s
                Adds the Command "Auto" to the CommandWindow that when selected will
batch process all selected files.
```

1.18 BarFormat

```
-------------------------------------------------------------------------------
                                  BarFormat
-------------------------------------------------------------------------------

Function: Specifies which information and in what order it is to appear
          in the title bar of the Workbench or DiskMaster screen.

Syntax: BarFormat [text][%V][%C][%F][%P][%T][%D][%M][%Y][%W]

Arguments:

text: Any text you like to be displayed in the title-bar.
%V: Displays the current version-number of DM.
%C: Amount of available CHIP RAM.
%F: Amount of available FAST RAM.
%P: Amount of available (public) RAM in total.
%T: Displays the current time.
%D: Displays the current date.
%M: Displays the current month (abreviated).
%Y: Displays the current year.
%W: Displays the current weekday (abreviated).

Example:

 BarFormat "DM %V Chip:%C  Fast:%F  Total:%P  %W - %D.%M.%Y  %T"

Will result in a title bar that might look like this:

 DM 2.2b9 Chip:1471496  Fast:13396056  Total:14867552  Tue - 22.Apr.97  22:12
```

1.19 Batch

```
                ---------------------------------------------------------------------------

                                  Batch
------------------------------------------------------------------------------

Function: Executes a DiskMaster batch file.

Syntax: Batch [
                file
                ]

Example:

 Batch
                %s
                Executes the selected files in the source window as DiskMaster  ↩
                   script files.
```

1.20 Button

```
                ---------------------------------------------------------------------------

                                  Button
------------------------------------------------------------------------------

Function: Assigns a function to the S (source) and D (destination) gadgets.
          Clicking on S or D gadget will invoke the command.

Syntax: Button "
                command
                "

Example:

 Button "Parent"

When the S or D gadget is selected, the "Parent" function is performed.
```

1.21 Check

```
                ---------------------------------------------------------------------------

                                  Check
------------------------------------------------------------------------------

Function: Totals the number of bytes in the specified files or path and
          determines wheter or not there is room on the destination device to
          hold the files if copied.
          If there is enough room, the amount of storage space
          that will be left is displayed.
          If there is not enough room, you will be informed.
          All feedback is displayed in the DiskMaster screen title bar.

Syntax: Check <path>

Arguments:

path: The AmigaDOS path which is to be totaled and
      compared to the destination.

Example:

 Check
                %s
                Totals the byte count of the selected files and directories and ←
                   indicates
wheter or not they will fit in the destination directory.
```

1.22 ChgCmd

```
------------------------------------------------------------------------------
                              ChangeCommand
------------------------------------------------------------------------------

Function: Requests that you select a command to be altered,
          then open a requester allowing you to edit the command
          without editing the configuration file directly.

Syntax: ChgCmd

Arguments: None
```

1.23 CloseWindow

```
------------------------------------------------------------------------------
                                CloseWindow
------------------------------------------------------------------------------

Function: Closes the current window.

Syntax: CloseWindow [number]

Arguments:

number: The number of the window to close.

Examples:

 CloseWindow
```

Closes the active window.

```
 CloseWindow 5
```

Closes the fifth window.

* Could be of use in ARexx-Scripts.

1.24 Color

```
                ------------------------------------------------------------------------
                                        Color
------------------------------------------------------------------------------
```

Function: Sets the RGB values (in hexadecimal) of each color register.

Syntax: Color [colors]

Arguments:

```
(no arg): Opens the palette requester allowing you
          to visually change the color palette.

colors: The set of hexadecimal values specifying the RGB value
        of the color palette register. The color registers are set from
        left to right with each RGB value separated with a space.
```

Examples:

```
 Color
```

Opens the palette requester.

```
 Color 999 222 EEE 358
```

Sets the first four colors.

* Notice the 4 Bit per entry values.

This function utilizes
ReqTools
when available!

1.25 Comment

```
                ------------------------------------------------------------------------
                             Comment (
                Recursive
                )
```

```
------------------------------------------------------------------------------

Function: Changes the comment text of the specified file.

Syntax: Comment <
                file
                > ["text"]

Arguments:

(no arg): Opens a requester with the current comment to be edited.

Examples:

 Comment
                %s
                Opens a requester with the current comment to be edited
for each selected file.

 Comment
                %s
                 "Babelfish`n`Chips"

Sets the comment "Babelfish`n`Chips" for each selected file.
```

1.26 Confirm

```
                ------------------------------------------------------------------------------

                                  Confirm
------------------------------------------------------------------------------

Function: Opens a requester displaying a message, a "Continue" gadget,
          a "Stop" Gadget and a text line.
          All text in this requester can be customized.
          If the user selects the "Continue" gadget the requester will
          dissappear and the remainder of the command string
          will be executed.
          If the User selects the "Stop" gadget, the requester will disappear
          and the remainder of the command string will be disregarded.

Syntax: <"text"> [Continue] [Stop] [edit text]

Arguments:

text: The text that will appear as headline in the requester.
      If the text is to contain spaces, it must be included in quotes.

Continue: The text that will be displayed in the "Continue" gadget.

Stop: The text that will be displayed in the "Stop" gadget.
      If this text is specified, the "Continue text must also be specified.

edit text: The text that will be displayed in the string gadget.
```

If any text is to contain spaces, it must be included in quotes.

Examples:

```
 Confirm " Drop dead?" "Yo!" "No way!";Quit
```

Opens a simple requester without string gadget to confirm the action "quit".

```
 Confirm "Name of encoded file?" "Go!" "Stop!" .uu;Extern uuencodex
                %s

                %r
```

Opens a requester which string gadget contains the string ".uu" that can be modified by the user and is accessed as
%r
.

1.27 Copy

```
                ---------------------------------------------------------------------------

                            Copy (
                Recursive
                )
-------------------------------------------------------------------------------
```

Function: Copies the specified file(s) to the specified destination.

```
Syntax: Copy <
                file
                > <
                dest
                > [NEWER] [RENAME]
```

Arguments:

```
NEWER: If this flag is set files will only be copied if the source file
       of the same name is newer than the file in the destination directory
       or if the file does not exist there.

Rename: If this flag is set, a requester will ask for the name
        that is to be set for the copy of the file.
```

Examples:

```
 Copy
                %s

                %d
```

Copies selected source files to the destination directory.

```
 Copy DF0: DF1:
```

Copies all files from DF0: to DF1:

Copy
%s

%p
Will clone the source file adding a ".bak" extension.

1.28 Delete

Delete

Function: Deletes the specified file(s).

Syntax: Delete <
file
>

Example:

Delete
%s
Deletes selected files and directories in the source window.

1.29 Deselect

Deselect

Function: Deselects files and directories based on pattern matching.

Syntax: Deselect [
pattern
]

Arguments:

(no arg): Opens a requester asking for a pattern to match.

Examples:

Deselect

Opens a requester asking for a pattern to match.

```
 Deselect *

Deselects all files and directories in the current window.

 Deselect *.info

Deselects all files and directories ending with ".info".
```

1.30 Dirlist

```
------------------------------------------------------------------------------
                              Dirlist (ARexx only)
------------------------------------------------------------------------------

Function: Gets the contents of the current window with
          selected / deselected status.

Syntax: Dirlist [variable]

Arguments:

variable: Name of the list variable to be used by the ARexx Program for
          the information obtained from the DiskMaster window.

Example:

address DM
Dirlist List
do i=1 to List.0
 parse var List.i name +30 size +8 time +9 date +7 prot +9 cmt +80 dir0 +1 sel
 say name size time date prot dir sel
end

Displays directory of the current window in console window.
```

1.31 Expand

```
------------------------------------------------------------------------------
                                  Expand
------------------------------------------------------------------------------

Function: Specifies whether the expanded AmigaDOS volume and path names or
          unexpanded device names will be used to display directory paths.

Syntax: Expand [ON|OFF]

Arguments:

(no arg): Toggles between the two modes.

ON: Specifies that all paths are to be expanded to their fullest.
```

OFF: Specifies that physical and logical device names will be used as
     part of the directory path.

Example:

```
 Expand ON
```

Tells DiskMaster to expand all path names.

* I prefer "Expand OFF" to prevent any troubles from equal volume names
  when I have the harddisk from a friend attached to my system.

1.32 Extern

```
                ----------------------------------------------------------------------------
                            Extern (
                Recursive
                )
------------------------------------------------------------------------------
```

Function: Use external programs as DiskMaster functions.

```
Syntax: Extern [S=value] [
                command
                ] [args]
```

Arguments:

```
S=value: This optional argument determines the stack-size for the
         externally executed programm.
         >value< is decimal and defaults to 4000.
         This argument MUST be the first or is otherwise
         interpreted as part of the [args].

args: Arguments normally used by the external program. The "
                %s
                " and
      "
                %d
                " flags may be used to specify source and destination directories,
       or selected source files or directories.

       To run multiple external functions from the same command,
       separate them by placing a "\" between each function.
```

Examples:

```
 Extern munpack
                %s
                 -f >nil:
```

Runs the "MUnpack" command providing the first selected file in the source
window as the name of the file to be "de-mimed".

```
 Extern s=30000 Tools:GfxCon %s Format GIF
```

Executes the programm "GFXCon" with a stack of 30000 bytes.

1.33 Find

```
                ------------------------------------------------------------------------------

                           Find (
                Recursive
                )
------------------------------------------------------------------------------
```

Function: Searches the path for files containing the current pattern.
When a file is found which matches the specified pattern, a requester
will appear asking if you want to continue the search,
abort the search or open a window displaying the directory
where the file was found.

```
Syntax: Find [
                file
                ]
```

Examples:

```
 Find
                %s
                Search the selected paths using the current pattern.

 SetPattern "*.txt"; Find DF0:
```

Search DF0: for files ending with ".txt".

1.34 Font

```
                ------------------------------------------------------------------------------

                                  Font
------------------------------------------------------------------------------
```

Function: Changes the font to be used for file and command display.

Syntax: Font [name/size]

Arguments:

```
name/size: The name and size of the font.
           The font must be located in the "Fonts:" directory.
           If the font is not found, the default font is used.
```

Examples:

```
 Font clean/8
```

Set font to clean.font, 8 point

```
 Font
                %s
                Set font to selected file ( i.e. SYS:Fonts/topaz/8, or Fonts:topaz.font)
```

Note: The use of proportional fonts is NOT recommended as they
will cause alignment problems!

1.35 HostScreen

```
------------------------------------------------------------------------------
                                HostScreen
------------------------------------------------------------------------------
```

Function: Use a puplic screen as host.

Syntax: HostScreen [name]

Arguments:

name: The name of a public screen.

Example:

```
 HostScreen BAUD
```

All windows after this will open on the BaudBandit screen.

1.36 Lock

```
------------------------------------------------------------------------------
                                 Lock
------------------------------------------------------------------------------
```

Function: Locks a window as either a source or destination window.
This function allows for the use of multiple source
and/or destination windows.

Syntax: Lock [S|D]

Arguments:

S: Locks window as source window.

```
D: Locks window as destination window.

Examples:

 Lock S

Locks last selected window as a SOURCE window.

 Lock D

Locks last selected window as a DESTINATION window.
```

1.37 MakeDir

```
-------------------------------------------------------------------------------
                                 MakeDir
-------------------------------------------------------------------------------

Function: Creates a new directory.

Syntax: MakeDir [name]

Arguments:

(no arg): Opens a string requester.

name: The name to be given to the new directory.

Examples:

 MakeDir

Opens a requester asking for the name of the new directory.

 MakeDir RAM:T

Makes a directory in RAM: called "T".
```

1.38 Move

```
                ---------------------------------------------------------------

                             Move (
                Recursive
                )
-------------------------------------------------------------------------------

Function: Copies the specified file(s) from the source to the
          destination then deletes the specified file(s) from the source.

Syntax: Move <
                file
                > <
                dest
                > [NEWER] [RENAME]
```

Arguments:

NEWER: This flag specifies that only files of like names which
       have a newer file date than the existing files are to be moved.

RENAME: If this flag is set, a requester will ask for a new name
        for the file that is to be moved.

Examples:

```
 Move
                %s

                %d
                Moves selected files from the source window to the destination window.

 Move
                %s
                 RAM:
```

Moves selected files from the source window to the RAM: directory.

```
 Move
                %s
                Requests the destination directory, then moves selected files
from the source window to that destination.
```

1.39 Msg

```
------------------------------------------------------------------------------
                                  Msg
------------------------------------------------------------------------------
```

Function: Displays text in the DiskMaster screen title bar.

Syntax: Msg "text"

Arguments:

text: Any text which is to be displayed in the DiskMaster screen title bar.

Example:

```
 Msg "Do something!"
```

Displays the message "Do something!" in the DiskMaster screen title bar.

1.40 NewDir

```
------------------------------------------------------------------------------
                                  NewDir
------------------------------------------------------------------------------
```

Function: Changes the directory path of the source window.

Syntax: NewDir [path|dmfile]

Arguments:

path: AmigaDOS path name.

dmfile: DiskMaster script file to be executed in lieu of reading
        a new directory.

Examples:

```
 NewDir
```

Changes the path of the current window to "none" which displays
the device list.

```
 NewDir RAM:*.info
```

Changes the path of the current window to RAM: (displays only .info files).

```
 NewDir %p
```

Refreshes the current window by selecting the current path as new path.

1.41 NewScreen

```
                ---------------------------------------------------------------------------

                                NewScreen
------------------------------------------------------------------------------
```

Function: Opens a new screen for DM.

Syntax: NewScreen [ID=value|D=value|H=value|W=value]

Arguments:

(no arg): Will bring up a screenmode-requester.

ID=value: The mode-id of the screen, you want DM to open.

D=value: The number of bitplanes to use for the custom screen.

H=value: The height in pixels to use for the custom screen.

W=value: The width in pixels to use for the custom screen.

All values are decimal.
Use utilities like aminet/util/misc/gmi22_usr.lha to obtain
the right mode-id for your prefered screenmode.
Or simply use
SaveConfig
to obtain
the perfect command-line for your screen.

Examples:

```
 NewScreen
```

A screenmode-requester will open from wich you may select your prefered mode,
only works when starting DM2 so far.

```
 NewScreen ID=167936
```

DM2 will open a PAL/Highres screen with the depth/height/width of your WB.

```
 NewScreen D=2
```

DM2 will clone your WB except from the number of bitplanes that is set to 2.

```
 NewScreen H=10 W=10
```

DM2 will open a screen with the mode and depth of your wb
and a size of 320 * 200.

1.42 OpenScreen

```
                ---------------------------------------------------------------------------

                                  OpenScreen
------------------------------------------------------------------------------
```

Function: Opens a custom screen.

This function is obsolete now, use
NewScreen
instead!

And from 2.2b12 the usage of OpenScreen will result in a
screenmode-requester popping up, just save your Startup.DM from
within DM to correct things!

1.43 OpenWindow

```
------------------------------------------------------------------------------
                                   OpenWindow
```

```
------------------------------------------------------------------------------

Function: Opens a file display or command window on the current DM screen.

Syntax: OpenWindow [left top width height [path|dmfile|CMD]]

Arguments:

left: The pixel coordinate for the left boundary of the window.

top:  The pixel coordinate for the top boundary of the window.

width: The width of the window (in pixels).

height: The height of the window (in pixels).

path: The DOS path to be read and displayed in the newly opened window.

dmfile:  The file name of a DiskMaster script file that will be
         executed upon the opening of the new window.

CMD: Command Flag.  This flag specifies that the newly opened
     window will take over as the current command window.

Examples:

 OpenWindow 100 150 200 50 RAM:

Open a new window which will load and display the directory "RAM:".

 OpenWindow 100 150 200 50 C:s*

Opens a new window which will display only files which start with
"S" in the C: directory.

 OpenWindow 100 150 200 50

Opens a new window which will display the device list.

 OpenWindow 100 150 200 50 CMD

Opens a new window which will take over as a new command window.
```

1.44 Pack

```
                ---------------------------------------------------------------------------

                             Pack (
                Recursive
                )
------------------------------------------------------------------------------

Function: Compress a file using internal single file compression method.
          A ".dmp" extension is added automatically.
```

```
Syntax: Pack [
                file
                ] [signature]
```

Arguments:

```
signature: An up to four character signature that will be included in the
           header of the file for later identification by an auto command.
           The pattern is: DMP0xxxx
```

Examples:

```
 Pack
                %s
                Opens a requester, asking for a signature then packs all selected files
using the same signature.

 Pack
                %s
                 txt
```

Using "txt" as signature will result in "DMPtxt" as header of the file.

* Nearly undocumented features are my favourite to explain... :-)

1.45 Parent

```
------------------------------------------------------------------------------
                                  Parent
------------------------------------------------------------------------------
```

```
Function: Displays the contents of the parent directory in the
          current directory path of the source window.
```

Syntax: Parent

Arguments: None

1.46 Pens

```
------------------------------------------------------------------------------
                                  Pens
------------------------------------------------------------------------------
```

```
Function: Allows you to customize which color register is used
          for each AmigaDOS pen.
```

Syntax: Pens [color]

Arguments:

```
color: This is the DiskMaster palette register number to be used
       for the specific AmigaDOS pen. There may be as few as two,
       and as many as eight register numbers, depending on the number of
       bitplanes being used.

       The position of the color numbers specifies which AmigaDOS
       pen will be assigned that color.  The positions are as follows:

        0 = Detail Pen
        1 = Block Pen
        2 = Text Pen
        3 = Shine Pen
        4 = Shadow Pen
        5 = HighFill Pen
        6 = HighFill Text Pen
        7 = Background Pen
        8 = Highlight Text Pen
        9 = Bar Detail Pen
       10 = Bar Block Pen
       11 = Bar Trim Pen - Only of use in V39 and up systems.

       Note: Refer to your AmigaDOS manual for a description of the
       purpose of each of these pen attributes.

Example:

Pens 0 1 1 2 1 3 1 0 3 1 2 1

Assigns color registers 0-3 to AmigaDOS pens, as specified by
their position.
This is the default setting.
```

1.47 Print

```
                -----------------------------------------------------------------------

                                  Print
------------------------------------------------------------------------------

Function: Prints specified file(s) (text or picture).

Syntax: Print <
                file
                > [HEX]

Arguments:

HEX: Specifies that the file is to be printed in hexadecimal
     format rather than in text or graphic format.

Example:

 Print
                %s
                Prints all selected files in the source window.
```

1.48 PrintDir

```
                ----------------------------------------------------------------------------
                                  PrintDir
------------------------------------------------------------------------------

Function: Prints selected files/dirs in source window using current format.

Syntax: PrintDir
                file
                 output

Arguments:

output: The logical device the text shall be passed to.

Examples:

 PrintDir
                %s
                 PRT:

Prints a list of all selected files using the printer.

 PrinDir
                %s
                 RAM:x

Generates a file ram:x that contains the list of selected files.
```

1.49 Priority

```
------------------------------------------------------------------------------
                                 Priority
------------------------------------------------------------------------------

Function: Change the current AmigaDOS priority of DiskMaster.
          Changing this priority value can have an effect on other running
          programs.  It is recommended that unless you are familiar with the
          effects of changing program priorities, you should leave the
          DiskMaster priority at its default setting.

Syntax: Priority [#]

Arguments:

(no arg):  Selecting this command will open a requester,
```

allowing you to edit the current priority value.

#: The new priority value.

Example:

```
 Priority 0
```

Sets the DiskMaster priority to 0.

* Totally unnecessary.

1.50 Protect

```
                ---------------------------------------------------------------------------------------
                            Protect (
                Recursive
                )
------------------------------------------------------------------------------
```

Function: Changes protection/attribute bits on specified file(s).

```
Syntax: Protect <
                file
                > ["[+] [-] [hsparwed] [G]"]
```

Arguments:

+: Specifies that the following bits are to be turned ON.

-: Specifies that the following bits are to be turned OFF.

```
hsparwed: Optional list of bits to be turned ON or OFF when
          preceded by a plus (+) sign or minus (-) sign.
          If the plus or minus signs are NOT used,
          the list will specify the absolute bit settings.
          The bits have the following meanings:

          H - Indicates that the file should be Hidden.

          * Ofcourse NOT!
            From OS2 to 3.0 "H" was the "Hold"-Bit,
            used by the OS to automatically make "pure" commands resident.
            The support for "Hold" was removed in OS3.1 .
            Please make no use of it.

          S - Indicates that the file is an AmigaDOS Script.
          P - Indicates that the file is Pure and can be made Resident.
          A - Archive Bit; used by hard disk backup programs to indicate
              whether or not the file has been backed up.  This bit is
              automatically reset to OFF if any other tool has written to
              the file.
          R - Indicates that the file may be Read.
          W - Indicates that the file may be Written to.
          E - Indicates that the file may be Executed.
          D - Indicates that the file may be Deleted.

          Note: If bits to be turned on or off are not specified, a
          requester will appear displaying the current protect bit
          settings, allowing you to edit them.

G: Global flag.
   When this flag is present, the requester will
   not appear for each selected file.
   Instead, the current bit setting are used for all selected file(s).

Examples:

 Protect
               %s
               Opens a requester, asking for protection bits to be edited for
each selected file.

 Protect
               %s
                RWED

Sets each file to "----RWED"

 Protect
               %s
                "+S-P"

Turns the script bit ON and turns the pure bit OFF on each selected file.

 Protect
               %s
                +D

Make all selected files deletable.
```

1.51 Quit

```
------------------------------------------------------------------------------
                                  Quit
------------------------------------------------------------------------------

Function: Exits DiskMaster.

Syntax: Quit

Arguments: None
```

1.52 Read

```
                 ---------------------------------------------------------------------------
                                  Read (
                 Recursive
                 )
--------------------------------------------------------------------------------
```

Function: Opens DiskMaster’s built-in file reader
to read the specified file(s).

```
Syntax: Read  <
                file
                > [HEX | Screen_Options | SEARCH["Text"]]
```

Arguments:

HEX: Specifies that the file reader is to use its "Hexadecimal" mode.
In this mode, the file will be displayed in both Hexidecimal and ASCII.

Screen_Options: These are the options for the screen the reader the will open,
compatible with
NewScreen
.
In fact, Read will call NewScreen with
a "d=2" as default value.

SEARCH: Specifies that a search is to be made on the specified file(s).

Text: If a search is to take place, specifies the search string.
If SEARCH is specified, but this argument is omitted,
a requester will open first to ask for the search string.

Using the Reader:

Once the reader has loaded the file and opened its screen,
the first page of text is displayed.
The file reader does not wrap the text.

Scrolling:

Cursor Left/Right (Keypad 4/6): Move the display window by eight columns.
Cursor Up/Down (Keypad 8/2): Move the display window by one line.
Shift Cursor Up/Down , Space/Backspace (Keypad 9/3): Move the display window page-wise.
T / Keypad 7 : Top of file.
B / Keypad 1 : Bottom of file.
Left Mouse : Toggle "Jump" mode / "Smooth" mode.

When in "Smooth" mode, placing the pointer in the center of the screen will
pause the scroll. Moving the pointer a short distance up or down from center
will cause the display to move slowly in the direction of the pointer movement.
Moving the pointer up or down to the near top or bottom of the screen will
cause the display to quickly scroll in the direction of the mouse movement.

Searching:

S : Opens a requester asking you for a string to search.

The search string requester will accept the wild card characters "?" (single unknown) and "*" (multiple unknown). When in hexadecimal display mode, entering a hexadecimal value preceded with a "$" will search for that hexadecimal string.

After entering the search string, the reader will search the file for the specified string. If the string is found, the portion of the file will be displayed where the string first occurs, with each occurrence of the string highlighted.

C/N : Continue the search.

M : Switch between HEX and ASCII modes.

To exit the file reader you can click on the close gadget at the upper left corner of the screen, or press [Q], [Esc], [Enter], or [Return], or the mouse.

Examples:

```
 Read
               %s
                w=640
```

Read selected text file(s) using an 640 pixels wide screen.

```
 Read
               %s
                Hex d=4
```

Opens the reader in hexadecimal display mode with a depth of 4.

```
 Read
               %s
                SEARCH "bubblegum"
```

Read selected text file(s) and search for the string "bubblegum".

```
 Read
               %s
                h=300 SEARCH
```

Read selected files, ask for a searchstring and open a 300 pixels high screen.

```
 Read
               %s
                SEARCH h=300
```

Won‘t work properly since DM assumes that "h=300" is your searchtext!

1.53 Recursion

```
                ---------------------------------------------------------------------------

                                Recursion
------------------------------------------------------------------------------

Function: Enables or disables recursion, allowing for functions
          to be performed on a directory without having it performed on the
          files within that directory.

Syntax: Recursion ON|OFF

Arguments:

ON: Enables recursion.

OFF: Disables recursion.

Example:

 Recursion OFF;Rename
                %s
                This will allow you to rename a directory, and will not attempt
to rename any files within that directory.
```

1.54 Rename

```
                ---------------------------------------------------------------------------

                                 Rename
------------------------------------------------------------------------------

Function: Renames specified files and/or directories.

Syntax: Rename <
                file
                > [new]

Arguments:

new: The file's or directory's new name.
     Without this argument, a requester will open asking for the new name.

Example:

 Rename
                %s
                Opens a requester asking for the new name of each selected file.
The requester uses the old name as a default, allowing you to edit it.
```

1.55 ReqPattern

```
                ---------------------------------------------------------------------------

                                ReqPattern
---------------------------------------------------------------------------------

Function:  Opens a requester asking for a pattern to be matched
           when directories are selected.  A default pattern, and custom
           text in the title bar and gadgets may be included.

Syntax: ReqPattern "text" [
                pattern
                ] [Okay Abort Cancel]

Arguments:

text: Text to be displayed in the requester’s title bar.
      If the text included spaces, it must be enclosed in quotes.

Okay Abort Cancel: This requester is used for both recursive and
                   nonrecursive operations, and contains three gadgets.
                   The first gadget accepts the pattern and closes the
                   requester, the second gadget aborts the entire (recursive)
                   operation and the third aborts the current operation.
                   The text for each of the gadgets can be changed,
                   however their functions will remain the same.

Example:

 ReqPattern "Please enter pattern" *.c

Requests pattern to be used in the current operation with "*.c" as the default.
```

1.56 Reset

```
---------------------------------------------------------------------------------
                                 Reset
---------------------------------------------------------------------------------

Function: Resets DiskMaster to a primitive state
          in which it contains no windows or menus.
          By placing this command at the beginning of a script,
          a running DiskMaster can be completely re-configured.

Syntax: Reset

Arguments: None
```

1.57 REXX

```
                ---------------------------------------------------------------------------

                                 REXX
------------------------------------------------------------------------------

Function: Executes the specified ARexx program or string.

Syntax: REXX [file|"string"]

Arguments:

file: Name of ARexx program to be executed.

string: ARexx command(s) to be executed.  The ARexx string must
be enclosed in quotes.

Examples:

 REXX "parse arg x,say x"
                %s
                Echoes each selected filename to the console window previously
opened with the "StdIO" command.

 Rexx FastDMView.rexx
                %p
                Executes the Arexx-Script "FastDMView" providing the current path
                   as argument.
```

1.58 Root

```
------------------------------------------------------------------------------
                                     Root
------------------------------------------------------------------------------

Function: Displays the root directory of the current window path.

Syntax: Root

Arguments: None
```

1.59 SaveConfig

```
                ---------------------------------------------------------------------------

                                  SaveConfig
------------------------------------------------------------------------------

Function: Saves a batch file that will recreate the current
          environment exactly.
```

```
Syntax: SaveConfig [
                file
                ] [ICON]

Arguments:

(no arg): Opens a requester asking for the name of the file to
          be written or overwritten with the environment information.
          You have 128 bytes in the string-gadget to enter
          the full path and name.
          The file will be saved with an icon when you press the
          "Icon" gaget in the requester instead of "Continue".

file: Name of file to be written or overwritten with the
      environment information.

ICON: Specifies that an icon is to be provided for the saved file.
               Has to be the second argument and upper-case!

Example:

 SaveConfig Startup.DM ICON

Saves the file as "Startup.DM" (with an icon).

 SaveConfig

A requester is asking you for a name of the Startup.DM that is to be saved.
```

1.60 ScrBack

```
-------------------------------------------------------------------------------
                                 ScrBack
-------------------------------------------------------------------------------

Function: Pushes the DiskMaster screen behind other existing screens.

Syntax: ScrBack

Arguments: None
```

1.61 ScrFront

```
-------------------------------------------------------------------------------
                                  ScrFront
-------------------------------------------------------------------------------

Function: Brings the DiskMaster screen to the front of other existing screens.

Syntax: ScrFront
```

Arguments: None

1.62 Select

```
                ---------------------------------------------------------------------------

                                  Select
-------------------------------------------------------------------------------
```

Function: Selects file in the source window using the specified pattern(s).

```
Syntax: Select [
                pattern
                ] [pattern2] ... [pattern29]
```

Arguments:

(no arg): Opens a requester asking for the pattern for the selection.

Examples:

```
 Select *
```

Selects all files and directories in the current window.

```
 Select *.info
```

Selects all files and directories in the current window which end with ".info".

```
 Select *.info *.readme *.nfo *.diz *.lha
```

Selects all files and directories in the current window which end with
".info", ".readme", ".nfo", ".diz" and ".lha".

1.63 SetDate

```
                ---------------------------------------------------------------------------

                             SetDate (
                Recursive
                )
-------------------------------------------------------------------------------
```

Function: Changes the file date and time of the specified file(s).

```
Syntax: SetDate [
                file
                ] [date]
```

Arguments:

```
date: The date to be written to the specified file(s).
      The date must be specified in the format hh:mm:ss ddmmyy.
      If no date is supplied, the current date and time will be used.

Example:

 SetDate
                %s
                Sets the date of all selected files in the source window to the
current time and date.
```

1.64 SetFormat

```
------------------------------------------------------------------------------
                                  SetFormat
------------------------------------------------------------------------------

Function: Specifies which, and in what order information is
          displayed in the file display windows.

Syntax: SetFormat [N] [S] [T] [D] [M] [Y] [W] [A] [C] [Text]

Arguments:

(no arg): Opens a requester allowing you to edit the current format.

N: File name
S: File size
T: Time
D: Date
M: Month
Y: Year
W: Day of week
A: File atributes (protection bits)
C: File comment

Text: Any text that is to be displayed along with the file information.

Examples:

 SetFormat

Opens a requester allowing you to edit the current format.

 SetFormat "S N A T W D-M-Y C"

Changes format to SIZE NAME ATTRIBUTES TIME WWW DD-MMM-YY COMMENT

 SetFormat "NS C"

Changes format to SIZE NAME COMMENT

* The "Text" option is somewhat useless since even qouted characters
```

```
  will be interpreted - text without a/c/d/m/n/s/t/w/y works fine...
```

1.65 SetList

```
                ---------------------------------------------------------------------------------

                               SetList (ARexx Only)
------------------------------------------------------------------------------

Function: Displays items in a list in a file display window,
          allowing you to use it as an ARexx item selection window.
          Using
                DirList
                , the status of this list
           (selected/deselected) is returned to the ARexx program.

Syntax: SetList <stem_name>

Arguments:

stem_name: The stem name of the list (defined in an ARexx program)
           to be displayed.  The list must be defined prior to
           issuing this command and must be in the following format:

           SList.1='One' SList.2='Two' SList.3-'Three'

Example:

 SetList SList

Displays the items in "SList" in the current file display window.
```

1.66 SetPattern

```
                ---------------------------------------------------------------------------------

                                SetPattern
------------------------------------------------------------------------------

Function: Sets the pattern for the current operation, overriding
          the pattern specified by "
                ReqPattern
                ".

Syntax: SetPattern
                pattern
                Example:

 SetPattern *.c

Sets pattern for current operation to *.c.
```

1.67 SetPrinter

```
------------------------------------------------------------------------------
                                SetPrinter
------------------------------------------------------------------------------

Function: Opens the printer option requester, allowing you to
          change the current printer setup.

Syntax: SetPrinter

Arguments: None

* Isn`t the requester-layout awful? This is annother subject to change.
```

1.68 Single

```
                ------------------------------------------------------------------------------

                                 Single
------------------------------------------------------------------------------

Function: This function tells the (
                recursive
                ) function to occur only once
          when activated, regardless of how many files are selected.

Syntax: Single

Arguments: None

Example:

 Single;Delete %s

Would delete only one selected file, without deleting any
directory or its contents.
```

1.69 Sort

```
------------------------------------------------------------------------------
                                  Sort
------------------------------------------------------------------------------

Function: Changes the sorting method.

Syntax: Sort [ N/0 | 1 | C/2 | 3 | 4 | D/5 | S/6 | 7 | F/8 ] [G]
```

```
Arguments:

N / 0: Sorts the active window by name.

  1  : Sorts the active window by name in reverse order.

C / 2: Sorts the active window by comment.

  3  : Sorts the active window by comment in reverse order.

  4  : Sorts the active window by date, oldest file first.

D / 5: Sorts the active window by date, newest file first.

S / 6: Sorts the active window by size, smallest file displayed first.

  7  : Sorts the active window by by size, biggest file displayed first.

In all these modes, the drawers are always on top of the lists.

  F  : Disables all sorting, added for testing purposes.

  G  : Global-Flag, resets all windows to the same sort-type.

Examples:

 Sort 0

The active window will be sorted by name.

 Sort 7

The active window will be sorted by size, biggest file first.

 Sort 4 g

All windows will be sorted by date, oldest file first.

 Sort g

All windows will be sorted by the last selected sort-type.
```

1.70 Status

```
-----------------------------------------------------------------------------
                              Status (ARexx Only)
-----------------------------------------------------------------------------

Function: Returns string in result for ARexx programs.

Syntax: Status <D|F|H|P|V|W>

Arguments:
```

D: Returns the current destination path.

F: Returns the current file display format.

H: Returns the host screen name or the name of the public screen
   that is identical with the port-name and the task-name.

P: Returns the current path of source window.

V: Returns "DiskMaster 2.2" or whatever the current version is.

W: Returns the number of currently active window.

Example:

```
 Status P
```

1.71 StdIO

```
------------------------------------------------------------------------------
                                 StdIO
------------------------------------------------------------------------------
```

Function: Opens or closes an Input/Output window to allow
          interaction with external program.

Syntax: StdIO [filespec] [CLOSE]

Arguments:

filespec: Standard console window information such as the
          console handler, window dimensions and window title.

Close: Closes previously opened I/O window.
       This argument may appear at the end of the command string which
       opened the I/O window.

Examples:

```
 StdIO "CON:240/20/260/40/DiskMaster I/O"
```

Opens a console window.

```
 StdIO CLOSE
```

Closes the window.

Note: The I/O window opened by DiskMaster is NOT a CLI window.
It is available to provide a method of program interaction for those programs
which such, but do not supply their own method of communication.

* This function will add an "/SCREENDM.n" automatically to your StdIO string
to make sure that the window will allways open on the right screen.

1.72 Swap

```
------------------------------------------------------------------------------
                                  Swap
------------------------------------------------------------------------------

Function: Swaps the directory paths between the two
          most recently active windows.
          If only two windows are open,
          will swap the directory paths between the two.

Syntax: Swap

Arguments: None
```

1.73 TitleFormat

```
------------------------------------------------------------------------------
                                TitleFormat
------------------------------------------------------------------------------

Function: Specifies which, and in what order information is
          displayed in the file display window title bar.

Syntax: TitleFormat [%C] [%F] [%I] [%B] [Text]

Arguments:

%C: Number of files in directory.

%F: Amount of available (free) space on the disk.

%I: Number of files selected in the window.

%B: Total number of bytes in the selected files.

Text: Any text that you may want to display in the window title bar.

Example:

 TitleFormat Selected: %I of %C Files totaling %B Bytes

Displays a mixture of text and system information resulting in a
window title bar that may appear as below:

Selected: 5 of 20 Files totaling 24674 Bytes
```

1.74 UnLock

```
------------------------------------------------------------------------------
                                 UnLock
------------------------------------------------------------------------------
```

Function: Unlocks the selected window, or all windows.

Syntax: UnLock [ALL]

Arguments:

(no arg): Unlocks the currently selected window.

ALL: Unlocks all windows.

1.75 UnMark

```
                ---------------------------------------------------------------------------

                                UnMark
------------------------------------------------------------------------------
```

```
Function: Enables or Disables automatic deselection of files during operations.
          This feature allows multiple operations to be performed
          on a set of selected files without the need of
          re-selecting them after each operation.
```

Syntax: UnMark [ON|OFF]

Arguments:

ON: Enables deselection.

OFF: Disables deselection.

Example:

```
 UnMark OFF;Copy
                %s

                %d
                Copies all selected files without deselecting them.
```

1.76 Unpack

```
                ---------------------------------------------------------------------------

                             Unpack (
                Recursive
                )
------------------------------------------------------------------------------
```

Function: Decompress a file using internal single file compression method.

Syntax: Unpack [

```
                file
                ]

Example:

 Unpack
                %s
                Unpack all selected .dmp files.
```

1.77 Wait

```
------------------------------------------------------------------------------
                                  Wait
------------------------------------------------------------------------------

Function: Causes a timed delay or a cued delay.

Syntax: Wait [seconds|text]

Arguments:

(no arg): Asks user to press [return] (A StdIO window must be open)

seconds:  The number of seconds to wait before proceeding with
          the remainder of the command string.

text: The text that will remind the user to press [return].

Examples:

 Wait 4

Waits for 4 seconds.

 Wait "Please press RETURN.....NOW!"

Waits untill the user has pressed the [return] key.
```

1.78 Window

```
------------------------------------------------------------------------------
                                Window
------------------------------------------------------------------------------

Function: Selects a window, making it the source window.

Syntax: Window <Path|Dest|Next>

Arguments:

Path: An AmigaDOS path.
```

The window that matches the specified path will be selected.

Dest: Selects the destination window.

Next: Selects the window which appears as the next window in the list.

1.79 These are gone now...

```
                ----------------------------------------------------------------------------

                      Removed functions since 2.1c:
------------------------------------------------------------------------------


                    DiskCopy

                     Format

                      Play

                  Backpattern

                    ShowPic
```

1.80 Play

```
                ----------------------------------------------------------------------------

                                  Play
------------------------------------------------------------------------------
```

Function: Play a music module.

```
Syntax: Play [
                file
                ]
```

Arguments:

(no arg): Stops the musik and unloads the module.

Examples:

```
 Play
                %s
                Plays the selected file as a SoundTracker/NoiseTracker/ProTracker ↩
                   module.

  Play
```

Stops playing.

```
* removed in 2.2b2
```

1.81 DiskCopy

```
------------------------------------------------------------------------------
                                DiskCopy
------------------------------------------------------------------------------

Function: Performs sector copy from source floppy disk to destination
          floppy disk(s).

Syntax: DiskCopy <source> <dest0> [dest1] [dest2] [dest3] [MULTI]

Arguments:

(no arg): Opens a diskcopy requester allowing you to visually
          set all diskcopy options.

source: The floppy drive to be used as the source drive.

dest0: The furst floppy drive to be used as the destination drive.

destN: Optional destination drives.

MULTI: Multiple Copy flag allowing more than one set of copies to be
       made from a single reading of the source disk.

Example:

 DiskCopy Df0: Df0: Df1: MULTI

Reads the disk in DF0: into memory, then wrties to DF0: and DF1: .
After each set pf copies is complete, you will be prompted to insert additional
disks into DF0: and DF1: to make additional copies.
Pressing ESC will abort the operation.

* removed in 2.2b2
```

1.82 Format

```
------------------------------------------------------------------------------
                                  Format
------------------------------------------------------------------------------

Function: Formats one or more floppy disks.

Syntax: Format <DFx:> [name] [DFx:] [name] ... [INSTALL|QUICK|FFS|VERIFY]

Arguments:
```

DFx:: The drive to be used to format a disk.

name: The name to be given to the disk being formatted in the
      previously specified drive.

INSTALL: Makes the newly formatted disk "bootable".

QUICK: Specifies that the directory of the disk will be cleared instead
       of a full format being performed.

FFS: Use the FastFilingSystem on that disk.

VERIFY: Specifies that the disk is to be verified during format.

Examples:

```
 Format
```

Opens DiskMaster‘s format requester.

```
 Format DF0: Empty0 DF1: Empty1 DF2: Empty2 INSTALL QUICK
```

Formats disks in DF0:, DF1: and DF2: with diff. names, installing each,
and uses the QUICK option (fast clear).

* removed in 2.2b2

1.83 Backpattern

```
                ---------------------------------------------------------------------------

                              Backpattern
------------------------------------------------------------------------------
```

Function: Loads any size IFF ILBM as a pattern.

```
Syntax: Backpattern <
                file
                > [[P]alette]
```

Arguments:

P: Use the color palette of the selected picture.

Examples:

```
 Back "Env:DMPattern.ilbm"
```

Loads the specified file.

```
 Back
                %s
                 P
```

Loads the selected file and uses it’s color palette.

Back

Clears the pattern.

* removed in 2.2b3

1.84 ShowPic

```
                ------------------------------------------------------------------------------
                              ShowPic (
                Recursive
                )
------------------------------------------------------------------------------
```

Function: Displays IFF ILBM picture files, or plays IFF 8SVX sound files.

```
Syntax: ShowPic [
                file
                ] [seconds]
```

Arguments:

```
seconds: The number of seconds the picture file is to be
         displayed (this argument does not affect sound files).
```

Example:

```
 ShowPic
                %s
                Shows/Plays all selected files in the source window.
```

* removed in 2.2b3

1.85 Flags

```
------------------------------------------------------------------------------
                              DiskMaster Flags
------------------------------------------------------------------------------
```

```
%a - The name of the ARexx-port / public-screen / task.
%d - The destination path.
%n - The name of a file or directory - disabled for 2.2b14
%p - The source path.
%r - The string from a requester. (also returned in RESULT for ARexx)
%s - The selected file or drawer with full path.
```

1.86 Command

```
------------------------------------------------------------------------------
                              The "Command" Keyword
------------------------------------------------------------------------------

command: The command-string DiskMaster will execute when you selected an
         option from the menu or the CommandWindow, when you double-clicked
         a file and it matched with one of the AutoCommands
         or when you press a key for a Key-Command.

The string may contain several internal or external functions seperated by ";".
```

1.87 File

```
                ---------------------------------------------------------------------------

                            The "File" Keyword
------------------------------------------------------------------------------

file: The file or drawer you selected to perform a function on.
      This could be provided absolute or as
                %s
                .
```

1.88 Pattern

```
------------------------------------------------------------------------------
                            The "Pattern" Keyword
------------------------------------------------------------------------------

pattern: Any printable text may be used as the pattern to be matched.
         The wild card characters "?" (single unknown) and "*" (multiple
         unknown) may also be used.
```

1.89 dest

```
                ---------------------------------------------------------------------------

                            The "dest" Keyword
------------------------------------------------------------------------------

dest: The destination path.
      The
                %d
                 flag may be used in place of this argument to specify
      that the path specified by the destination window is to be used.
```

1.90 Recursive

```
                ------------------------------------------------------------------------------

                                 Recursive
------------------------------------------------------------------------------

Functions marked as "Recursive" will work on all selected files and
all files in selected directories by being invoked untill the function
was performed on all of them.

When you have five files selected and perform the
                Comment
                 function
on them, then the requester will pop up five times to let you enter the comment
for the selected file in the row.

* The default-setting for "comment" is:

AddCMD Comment, Comment,Recurse OFF;Comment
                %s
                So you won't be able to set the comments for files within a  ↩
                   selected drawer.
Otherwise you are able to set a comment for a selected drawer.
```

1.91 The goodies included...

```
                ------------------------------------------------------------------------------

                You should find these files in the archive:
------------------------------------------------------------------------------


                C

                Icons

                Rexx
                DM2               58992 Bytes 2.3 - The actual binary.
DM2.guide        123098 Bytes 1.58 - The document you are currently reading.
DM2.guide.info     1576 Bytes - The Original Icon
DM2.info           1567 Bytes - A nice magic WB icon,
                                provided by Eddie Ivanov (eivanov@tfs.net)
MyStartup.dm       9337 Bytes 2.34

My very own Startup.DM provided as example on how DM2 could look like.

MyStartup.DM.info   850 Bytes - A non-lace Icon I created using the original.
```

1.92 Tiny but usefull...

```
------------------------------------------------------------------------------
                            Small little helpers:
------------------------------------------------------------------------------

Head   388 Bytes 1.0 - by Synthetic/Incal

Head:

This was written for me by a friend some years ago.
Usage: Head %s
Result: The first 2048 bytes of the given file are copied to ram:head
Purpose: Allows a fast sneak into binaries with DiskMaster
```

1.93 Look at these!

```
------------------------------------------------------------------------------
                              Alternative Icons:
------------------------------------------------------------------------------

DM2_Pat_Giever.info  1657 Bytes - Drawn by Pat Giever,
             I received this Icon from Clay Alberts <clay@umcc.umcc.umich.edu>
```

1.94 Arexx-Magic

```
------------------------------------------------------------------------------
                           ARexx-Scripts included:
------------------------------------------------------------------------------

AssignList.rexx     1723 Bytes     by Steven D. Kapplin
DM-Vir.DM           2765 Bytes 1.1 by Inorix Scripts (Jan-willem van Eys)
DMSpopper.rexx      1162 Bytes 1.0 by R.Riedel
FastDMView.rexx      775 Bytes 2.2 by R.Riedel
FileShow.rexx       9000 Bytes 3.1 by Inorix Scripts (Jan-willem van Eys)
Sub_FileRecog.REXX  3344 Bytes 2.8 by Inorix Scripts (Jan-willem van Eys)
WindowTest.rexx      246 Bytes 2.0 by R.Riedel

AssignList.rexx :

AssignList.rexx - DiskMaster utility to get assign an create new command
window which contains a listing of all mounted volumes and assigns.
When you click on any items in the list a new DM window will open
with a listing of the directory for the selected device or assign.
Makes it easier to navigate through you hard drive system.
You may also add additional devices which may not show up in
the lists produced by ARexx's showlist() function.

DM-Vir.DM :

Intelligent Viruskiller script for DiskMaster II.
It will check for running Virus_Checker v7.18+ or VirusZ II v1.33+.
```

If neither is running, it will start one.

DMSpopper.rexx :

An example script to show the power of ARexx.

It opens a new command window for special treatment of autodetected DMS files.

FastDMView.rexx :

A multi-selection script for FastView.
FastView has the ability to load and process as much pictures as there
is space in your memory.
The bad thing is that DM has no multi-selection feature and FastView‘s
command line is restricted to 256 Bytes in length.
So this script is the solution.

FileShow.rexx :

A File_ID.diz extractor for DiskMaster II

Sub_FileRecog.REXX :

FileRecog subroutine

FR_T clip : Filetype
FR_S clip : Short type identifier
FR_I clip : first 48 chars as hex string

```
 Known filetypes:
 ----------------
 Archives: LZH,LHA,LHA-SFX,LZX,ZIP,ARC,RAR,ARJ,DMS,Zoom,XMash,XArchive,
           PackDev.
 Packed  : XPK,PP20.
 Images  : GIF,JPEG,IFF,ANIM.
 Sound   : WAVE,8SVX.
 Other   : EXE,ICON,GUIDE,IntroCAD.
 IFF     : ILBM,ANIM,8SVX,CTLG,PREF,CDAF,FAXX,STFax.
```

WindowTest.rexx :

Just an example for the functions introduced in 2.2b8 .

Add "AddMenu Test, WinRexx, Rexx WindowTest.rexx %a" to your Startup.DM .

1.95 All the little helpers listed...

```
--------------------------------------------------------------------------------
      This tells more about all the little extra utilities beeing used:
--------------------------------------------------------------------------------
```

Most of the tools I use to expand the capabilities of DM
could be found on aminet.
Nevertheless I‘ll try to give you more information on all these little
helpers like the complete path on aminet and the version numbers.

```
FTPMount: Mounts FTP sites as part of a filesystem.
          Version: 1.0
          Path:    Aminet/comm/tcp/FTPMount_1_0.lha

HTMLess:  Converts HTML to standard text files.
          Version: 2.5
          Path:    Aminet/text/hyper/htmless.lha
          Usage:   Extern HTMLess -w %s

From Fredrik Ismyren:

ClearRAM: It  will  wipe all files and directories from RAM:
          (regardless  of  their  protection  state),  but
          leave  disk.info, ENV, T, Clipboards intact.
          Version: 1.01
          Path:    Aminet/util/misc/ClearRAM.lha

Den6:     Small readable 7x6 font for 640x256 WB.
          It's perfect for Diskmaster!
          Version: 1.0
          Path:    Aminet/text/font/den6.lzh
                   (Didn't find it in Scandinavia!?)
                   * Also available on Aminet CD 8.

Nuke:     This small program is a secure "delete" command.
          Secure means that the file is physically destroyed
          and cannot be recovered in any way.
          Version: 1.0
          Path:    Aminet
```

1.96 The places to go...

```
                ------------------------------------------------------------------------------

                 Here you will find new versions of DiskMaster:
------------------------------------------------------------------------------

Aminet/util/dir  - I will upload new versions, the bad thing for you is
                   that I have to wait ten days before I`m allowed to upload
                   a new version.

comp.sys.amiga.binary  - I gave my permission to distribute DM over this
                         newsgroup moderated by Bill Zimmer (zim@ibx.com)

http://www.on-luebeck.de/~rriedel - Online since 97-04-04, this webpage was
                                    designed by
                Richard Backhaus
                .

ftp://ftp.on-luebeck.de/pub/systems/amiga/diskmaster2
                                  - Online since 97-06-03 01:02:12

AMiGA CiTY bBS (Germany)
------------------------
```

```
  Netname: ACiTY
   DOMAIN: .DAME.DE
   Number: +49-[0]911/698527 [28k8/ISDN]
           +49-[0]911/698551 [33k6]
   Online: 24h
   System: A2ooo/6o
  DM-Area: /Support World/Diskmaster
 Username: DM
```

1.97 Bugs

```
------------------------------------------------------------------------------
                 These are the known Bugs in DiskMaster 2.1c
------------------------------------------------------------------------------

Please report any bugs to: DMBugs@amiga.on-luebeck.de!

I can`t answer all your mails anymore but I will react on them,
so please keep this mails as short as necessary, thank you.

And please try to be precise.
An "I think there once was a bug when I tried somethink like this..."
is not very helpfull.

- The Text Reader is limited to 10000 lines or 500k or something,
  misses the end on internally crunched files
  and scrolls to the next matching string while searching instead of jumping.

- The listviews can`t be scrolled using the keys.

- not 100% C -> even more work to make it portable to pOS or the like.

- Shows wrong amount of space on drives >2gig,
  reported by Georg Hazianastasiou (haz@geocities.com).

- Large list of files are scrolling if you selected one file at the bottom
  and copy it instead of jumping to this entry
  Same problem with "move", "printdir", "delete"...

- When you start DM without "NewScreen" in your Startup.DM,
  DM will open on the WB, this is OK.
  But when you select "NewScreen" from an window then,
  the screen will open but DM won`t move to it`s new home.

- Something like "%s.x" is ignored on the command line.
  Reported by Rainer Koppler (rk@gup.uni-linz.ac.at).

- PATH`s added with the PATH command during startup-sequence
  are not recognized by DM.
  Reported by Nils Goers (N.Goers@Art-line.de).

  This is no problem with DM.
  Workbench processes do not support the PATH, launch DM from
  a shell and it will work the way you expect it to.
```

- The dir-windows are not locked from access during scanning a new dir. Reported by Xavier Messersmith (xcaliber@wco.com).

- %n is only working correctly if there is also a %s within the command-line.

- Pressing "ESC" while DM is doing nothing but waiting will deactivate the "S"/"P" buttons untill the next command is executed. Reported by Andre' Vellori. Hey, looks like this bug is really old.

- "AddCmd MoveAS, 30, Move %s %d RENAME" will open a requester TWICE. Reported by Fredrik Ismyren (ismyren@hem1.passagen.se).

- "Window NEXT" seems to fail if there is only the CMD-Window left, reported by Hannu Nevalainen (henk@it.kth.se).

1.98 You want it?

What features do you like to see in DM?

Development will go on!

At first I want to fix some bugs and implement some of the features I had in mind for a long time now.

I like to get DM even smaller and faster, with more funtionality of course.

And this includes that I won‘t use MUI!!

But I will make use of new OS functions and therefore the support for Kickstart 1.3 has already been dropped!
And If there is something I just have to implement from OS 3, the support for OS 2 will also be sacrified.

And now it‘s your turn.
If you have any ideas on how DM could be improved, then send me a mail!
But please read the guide carefully before.

Some suggestions:

```
Localisation - Requested by Fabian Sloos (fsloos@hol.nl),
                            Georg Hazianastasiou (haz@geocities.com)
                            Dieter Kaye (MaxII@B-52.ohz.north.de).
```

I already thought about making all internal strings user-editable via a file called Strings.DM or something.
Real .locale support looks to unflexible to me.

Compare - Requested by Bill Simon (bills@webgate.net).

A function that checks two identical drawers if files have changed.

Purpose: Easy Backups using DM.

My suggestion: Write an ARexx script for this purpose.

Requested by Jody Tierney (jtierney@cyberlink-inc.com):

CTRL-X, CTRL-K, & CTRL-U)

This will be implemented nearly automatically when I remove the silly
requester stuff in DM using more decent system functions.
There is currently no Gadtools-Gadget in DM.

XPK support for pack/unpack

No, I don't like this idea, yet.
But I'm getting familiar with this...

Buffered Directory windows

DOS offers directory notification, so I may implement this
when I finally expand DM.
Or not, as I read in the autodocs that no aplication should
depend on StartNotify().
But then there must be a way to maintain a list of constantly
checked dirs even if there are no windows open for them.

Real Buttons

This would look better but then it takes more space on the screen,
more RAM:, more code - later.
One idea, has anyone implemented Toolmanager into DM using ARexx?

different Fonts for title, menus, window text...

Requested by Georg Hazianastasiou.

Later, let me fix some bugs first.
This will also lead to a new screen handling.

Sub-Menus - Requested by Chris Hall (hybrid@slinknet.com),
Dave Pierson (fishb@gte.net)

This will be part of pushing DM beyond OS 1.3...
The whole menu-routines have to be rewritten to make use of Gadtools.

Requested by Nils Goers (Goers@Art-line.de):

Font-Sensitivity

At the moment I see no pleasant way to implement this.
Each line in the Dir-Windows will have to be calculated individually
and this could be quiet time consuming.

-A flag for the name of the port

```
Implemented. (2.2b8)

HEXmode for data Recognition in AutoCMDs

Yes, I should include a "$" option on the commandline.

More Keys addressable - Requested by Manuel Martin-Vivaldi (manuel@webmedia.es)

And this is also a good idea, even though I'm a mouse fan.

Named windows - Requested by Xavier Messersmith (xcaliber@wco.com)

Implemented. (2.2b8)

NewLook Screen/Menus/Gadgets - Requested by all of you.

This is -under construction-!
But this is not easy as some of the routines should rather be rewritten
than improved as the most of the GUI code is 1.3 dependant.

non-resizable, non-dragable, and/or non-closeable  dir-windows
Requested by Kenneth "ALBANASINIA" Weese (alban@citynet.net) and Nils Goers.

I personally think that the free windows are one of the best features in DM.
But I may implement this when I re-write all the window-handling.

put DiskCopy/Format/Play/Backpattern/ShowPic back

DiskCopy - Why?
Format - Why? Floppy disks are dead media!
Play - Yes, I'm missing it and I think it will return once.
       But I need a system friendly and free source for a player first.
BackPattern - Totally useless cosmetics, if you like it colorful, DM is
              certainly not the right utility for you.
ShowPic - Same as Play but the only acceptable form to let it return is as
          system compliant datatypes viewer - and it will return...

Requested by too many people to just ignore their requests,
even though they are the minority.

Requested by T.L.Bullock <havoq@pandemonia.demon.co.uk>:

> Would be nice to have a "variable" for the first file selected in
> the destination window. This could be used for file merges, sorts etc.

After I checked for this - no problem.
But I also have something bigger in mind...
```

1.99 Look here first...

```
                ------------------------------------------------------------------------------

                                   Ask Dr.DM
------------------------------------------------------------------------------
```

I‘ll try to present this like a FAQ, answering to all kind of questions.

Q: Is there a "Delete *.info" command?

A: Use "Select *.Info;Delete %s"

Q: If you have a program such as Cybershow that can show multiple files
like this: "cybershow <options> file1 file2 file3 file4"
How may I use this with DM?

A: Check out FastDMview.rexx, this is taylor made for this purpose.
(And finally working 100% since 2.2)
Also read the hint in
Archive
.
A simple to use multi-selection feature is already planned.

Q: Is there an opposite to
SetFormat
, maybe "GetFormat"?

A: There is, use "
Status
F".

Q: Is there a way to let DM open it‘s windows on WB?

A: Just leave out the
NewScreen
command-line.

Q: Is there a way to have buffered directory-windows?

A: There is, open a new window, adjust it to the required path and
to minimal size, then throw it in the background.
All paths from opened windows will be checked periodically.
I just remembered that I used this technique with my old A500
and it‘s slow disk-drives.

Q: I like to start a script/launch some tools/execute an ARexx script
when starting or quitting DM, is there a way?

A: The answer is: Startup.DM !
Startup.DM is a batch that is interpreted line by line when DM starts.
So you may add any commands you like to it:

```
;$VER: Startup.DM 2.2...

Extern execute s:myDMstartscript
Rexx dosomethinkonDMstart.rexx
makedir ram:c
copy c: ram:c
Reset
```

Q: Everytime I start DM it comes up with the screenmode-requester...

A: You read this? Then try to read some of the rest...

1.100 All the tortures listed below...

```
                ---------------------------------------------------------------------------

                 What happened to the binary so far:
-------------------------------------------------------------------------------

2.3 (97-10-10) 58992 Bytes

- 'Status P' caused enforcer hits when there was no file-window - fixed.
- Each window is now sorted individually.
- 'Sort Size' works in both directions now.
- Sort called without arguments caused an enforcer-hit - fixed.
- 'Sort Name' in reverse order implemented.
- Added a 'Global' option to 'Sort'
- Thanks to Hans Guijt <H.Guijt@inter.NL.net> the proportional-gadgets
  appear in that 3D-newlook now, Greg only forgot a PROPBORDERLESS.
- Added a new function called 'NoDoubles', suggested by Hans Guijt.
- Set SA_SharePens = TRUE for DMOpenScreen() so DM will share the pens
  now instead of allocating them all. Greetings to Pontus Fuchs!
- 'Sort Comment' and 'Sort Date' in reverse order implemented.
- Added the new sorting modes to the internal config.
- Deleted four assembler routines that were leftovers from 'Show'.
  This saved 304 bytes.
- Fixed the enforcer-hit that occured when calling 'SaveConfig'
  without any arguments.
  Reported by Jeff Grimmett <Jeff_Grimmett@elric.maximumaccess.com> - Thanks!
- Fixed a bug in READ. The display-id was buffered as UWORD instead ULONG.
- FONT is using an ASL-Fontrequester now.
- Added size-checking to 'NoDoubles'.
- Fixed a bug in Copy NEWER that made this function useless.
  Reported by Wolfgang Ernicke - Thanks.

2.2b15 (97-09-24) 58464 Bytes

- A security check in the new DosExecute() function prevented the
  execution of external programms when there was no file-window.
  Thanks for the fast report to Fredrik Ismyren.
- Added a new security check to DosExecute().
  Thank you, Heinz Wrobel, for showing me that possible
  error-source.
- Optimized ParseScreenArgs() and DosExecute() by Elimination
  of a function call.
- Added the new sorter-sources from
                Richard
                :

  The new "Benchmarks":

       1422      2816      4909  Files

   00:08:11  00:27:00  01:14:90  2.1c   Sort by Name
   00:06:77  00:22:69  01:03:74  2.2b14 Sort by Name
```

```
    00:02:73  00:05:11  00:08:69  2.2b14 Sort by Fast (no sort)
    00:02:75  00:05:75  00:09:75  2.2b15 Sort by Name

  All in plain ANSI-C!
  Now we have to speed-up the seek...
- The new sorter is also handling special characters correctly now.
- Overall Cleanup of the source, created a new File called DMSort.c,
  dumped all the obsolete ASM-Source.
- Optimized the sorter by means of speed and size.
  10002 entries are sorted:
  Fast: 00:19:37  Name: 00:20:52  Date: 00:20:26  Size: 00:20:62
- Sorting by date is now done like the wb - newest first.
- Implemented a sort-by-comment.
- Re-worked the sort-by-date to have drawers on top of the list.
- Sort-by-comment caused enforcer hits, fixed,
  should be close to waterproof now.
- Fixed a bug that I introduced with 2.2b12.
  Copy NEWER also copied the file if it had the same date.
  Reported by David Brown <David.Brown-6@pp.ksc.nasa.gov>.
- Fixed SaveConfig() that I left unfinished in 2.2b12,
  now works like the original but without the bugs...
- Restructured parts of the source.
  All sorting is now done in one file.
  Made a new function out of three identical passages, this saved 150 bytes.
  Also increased the readability for later analysation.
- Fixed the bug that prevented the usage of the reader from two DM‘s
  at the same time - the pub-screen names were identical.
  The name of the reader-screen is now: "    DM-Reader.DMnumber"
  The white spaces are necessary for cosmetic reasons only.
- Fixed annother rarely occuring Enforcer-Hit that apeared since 2.04 .
  Thanks to Hannu Nevalainen again for reporting it!
- CompareDates() is now also used to determine if the FileInfoBlocks
  have been updated.
- Volumes are now renamed using Relabel() instead of sending an
  'ACTION_RENAME_DISK'-packet, this saved 112 bytes.
- Restructured the source by moving some functions around and
  re-formated them to increase readability.
- Further restructured and cleaned-up the code,
  removed some dead lines.
- Created a new source-file named DMRexx.c in which all Arexx stuff is done.
  The source is now splitted into 12 files.
- Inserted some checks to prevent a '//' in the dir-name.
  First reported by Bryan K. Williams <bryanw@connectnet.com> - thanks!

2.2b14 (97-08-24) 57788 Bytes

- Finally fixed an internal problem with the version-string.
- Started to implement an multi-select feature...
- Fixed a bug that caused Enforcer-Hits when no window was opened.
  This was reported by Hannu Nevalainen - thank you!
- Fixed a very stupid bug that only occured when starting DM
  without Startup.DM.
  Thanks to Thomas Kelly <tgkelly@umd.umich.edu> for reporting this!
- From 2.2b12 the internal reader was not able to open it‘s screen
  anymore when DM was running on the workbench.
  Fixed that and optimized the screen-routines.
  Thanks to Mark de Jong <dejong@tip.nl> for reporting this.
```

```
- Now the task-name is also set to DM.1 DM.2 DM.n ...
- Further optimized the screen-routines.
- CTRL-C will now work on programms that are started using "Extern".
  Thanks to Heinz Wrobel <heinz@hwg.muc.de> for helping me.
- Written a totally new function for %n, with the obvious problems...
- With the help of Martin Berndt <m_berndt@wanderer.gun.de> I was able
  to give you control over the menu colors.
- Recompiled with the new SAS-C 6.58
- Fixed a little bug in the command-line parser for "NewScreen".
- Rewritten the source for "Extern" totally.
  Added a new stack-size option as a little bonus.
- Removed the code for %n and %m in order to make this version
  at least distributeable - saved some bytes too. :-)
- Changed the "About"-requester text.

2.2b13 (97-07-13) 57980 Bytes

- There where two tables for hex-conversion, removed one.
- TAB won`t activate the string gadgets anymore but you are still able
  to fill the gadget with tab-chars as reported by Nils Goers.
- Fixed internal problem with the version-string.
- Implemented OS date-conversion and removed it due to massive
  problems with the layout, maybe later.
- As suggested by Jody Tierney on the mailing-list, the reader was expanded.
  Pressing 'N' will continue the search as well as 'C'.
- Fixed a very nasty bug that crashed the whole system when you called
  "Extern" or "Rexx" without any arguments.
  Thanks to Gregory Donner <gdonner@cyberlink-inc.com> for reporting
  the resulting enforcer-hit.
- Ten hours after I`ve released 2.2b13 I`m awake again.
  First think I noticed was the wrong version number within
  the screentitle...

2.2b12 (97-07-07) 57980 Bytes

- Oops, just remembered that the maximum amounts of arguments is 29 anyways
  and set this as maximum for the saved colors.
- Played around with "Pens" and found some interesting things.
  The pen-array will be expanded for new pens like "DirColor Pen" to make
  the selection GUI independant, the color of drawers is now set to pen 8,
  the color of selected entries is now set to pen 3.
- Also traced the problem with the black menu-bar.
- Tried some tricks to speed up the sorting and received a 30%
  faster sort "on the fly".
  Now it`s
                Richard`s
                 turn to make this water-proof...
- As suggested by Jan-willem van Eys in the mailing-list,
  the port- and screen-names are now: DM DM.1 ... DM.42 ...
- Changed the code to have most of the libraries auto-initalized.
  As a result, DM won`t fail silently any longer.
- Re-printed parts of the source.
- Modified both OpenScreen() functions a bit.
- With the help of Jody Tierney, I was able to fix that
  infamous "Fastview-Bug" with executing external programs.
- Fixed a bug with StdIO windows opening on the wrong screen.
- Fed up with questions, "OpenScreen" is now redirected to "NewScreen"
```

```
  without any arguments -> a screenmode-requester will pop up!
- Modified "Read" to make use of "NewScreen".
- Rewritten parts of "NewScreen" to eliminate an possible error.
- Disabled the source for "OpenScreen"
- Changed parser for "Read" to allow mixing of "Read" and
  "NewScreen" arguments for "Read" commands.
- Disabled a forgotten line from the code.
- Fixed the bug with saving the wrong icon.
- Corrected the icon-image structure by converting the icon:
  icon->.iff->c.source
  This saved 300 bytes again that where formerly wasted!
- Two bugs in "SaveConfig":
  - "Expand ON" after saving the Startup.DM.
  - The icon was always saved.
  Rewritten parts of the function.
- Optimized var handling saved a few hundred bytes from the exe.
- Changed the port- and screen names again: DM.1 DM.2 ... DM.42...
- Removed annother dead line from the code.
- RexxSysLib and ReqTools are the only libs left
  that are not auto-initialized.
- Tested with Enforcer, Mungwall and Segtracker still no hits.
- Tested with Stackmon, was not able to exceed stack-usage beyond 2300 bytes.
- Optimized "Copy NEWER" by usage of CompareDates().
- As suggested by Hannu Nevalainen in the mailing-list,
  SPACE / BACKSPACE are now used for PAGE DOWN/UP within the reader.
- Still not satisfied with SaveConfig(), a release is more important.
  Therefore also the new code for sort and pens were left out.

2.2b11 (97-05-19) 58808 Bytes

- The Sort functions are a bit optimzed and a "Sort Comment"
  function has been added that is not fully functional yet.
  In fact it wasn‘t meant to be integrated yet as there are several
  problems to solve with "sorting by comment".
  All done by my Co-Author
                Richard
                .
- ASL-Library integrated.
- Minor internal changes.
- Screenmode-Requester integrated into "NewScreen",
  examines the Workbench to determine the default data,
  will popup when no argument is given,
  time to rewrite the parse and save functions.
- "NewScreen" nearly waterproof now,
  argument parsing rewritten for this one.
- Fixed a little bug with "Rename" the string-gadget was too small.
  Reported by Timo Ronkko (deadbeat@iwn.fi).
- Disabled "Sort Comment" in order to release this version sooner.
- Set "SA_Overscan,OSCAN_TEXT" for OpenWindowTags() and removed
  "ASLSM_DoOverscanType,TRUE" from the screenmode-requester.
- Fixed a bug with the new sort-types "Fast" and "Comment",
  "Comment" still disabled though.
- Annother small speedup in sorting, only visible with 68000.
- The "NewScreen" screen is now using the "correct" Pens.
- Rewritten SaveScreen() to save the proper line for "NewScreen".
- Improved "OpenScreen" in order to make SaveScreen() work.
- Changed internal default Startup to call "NewScreen".
- A problem with the screen prevented the release in the last minute...
- ...and has been fixed now.
- Limited the maximum saved colors to 32,
  will be further reduced to the used pens in the future.

2.2b10 (97-05-01) 58300 Bytes

- Implemented a
               ReqTools
                color requester.
  When ReqTools is not available, the old requester will be used instead.
  DM2 has grown 92 bytes to make this possible. ;^)
- Fixed a little bug in "BarFormat", forgot to shorten the
  time-string after removing the AM/PM.
  Reported by Jody Tierney.
  Yes, I`m reading your mail!
- (97-04-29 23:30) found a way to nullify the startup delay,
  half an hour`s work is doing a great job but causes severe crashes when
  selecting a menu item...whole menu will be gadtools driven...
- Decided to release this version and therefore temporaly
  disabled some new portions of the code

2.2b9 (97-04-27) 58200 Bytes

- Again, some dead code removed - SaveConfig/BackPattern
- After reading a mail from Hannu Nevalainen (henk@it.kth.se),
  I instantly changed the clock to 24 hours format,
  saved again the huge amount of 36 bytes... :^).
  (I wanted to change this for a long time, just forgot about it)
- Fixed "About"
- Messed around with DM`s DOSExecute(),
  returned to the original function after some tries and
  added "SYS_UserShell = TRUE" to it.

2.2b8 (97-04-18) 58232 Bytes

- "Status D" is working again - some bugs are real stupid,
  at least I`m producing my own now - easier to fix...
- Just annother breakfast-break idea: "Select" is now multi-arg.
- Brought "Status W" to life.
- Implemented %a, this reflects the name of the arexx-port now.
- The name of the second ARexx port is now DM1, the third DM2...

2.2b7 (97-04-09) 58264 Bytes

- "NewScreen" improved, now accepts Depth, Width, Height, `S`, `P` ,
  basically clones the WB-Screen - work in progress...
- Implemented GadTools` LayoutMenuA(),
  DiskMaster went flaky, so it was disabled for the moment.
- "accidently" ran into parts of the commandline-parser,
  added %n on the fly.
- Fix: copy %s %s will no longer crash the machine.
- "Status W" should return the number of the active window,
  wasn`t able to reproduce this, as well as "Status D".
- CloseWindow [Number] implemented.

2.2b6 (97-04-05) 57952 Bytes
```

```
- Cloned "OpenScreen" Routine to "NewScreen" for testing purposes.
  Some portions of the code are really a mess.
  For example, Greg used OpenScreenTagList() and named the tags by their
  numbers instead of their names:
  Tags[t].ti_Tag=(1<<31)+0x32; == Tags[t].ti_Tag=SA_DisplayID;
  You may use "NewScreen" for a Test but it‘s really not finished,
  it will ignore all arguments so far.
- Removed some dead code again.
  Without Backpattern the Backdrop-Window was useless...
- Fixed a small bug in the default preferences.
  Reported by Donald Dalley (ab732@freenet.toronto.on.ca)
- Also removed ZOO and ARC lines from the default preferences.
- VBI Timer code Re-Integrated, was a severe mistake to remove it.
  DirWindows will Update again.
  CPU iddle is now dependant on how many windows you keep open,
  XOper reports now somethink between 1% and 4%.

2.2b5 (97-03-28) 58344 Bytes

- Received reports from Wolfgang Ernicke (kes132@freenet.mb.ca) and
  Timo Ronkko (deadbeat@iwn.fi) that DM‘s title screen is flickering and
  the cpu-iddle is to high when DM is waiting.
  Removed all VBI Timer code and modified main().
  This stopped the flickering which was caused by constantly updating
  the screentitle - that also caused the high CPU load.
  According to Xoper 2.6, the CPU usage dropped from 25% to 0% while
  DM is doing nothing but waiting.
  The drawback of this is that the title bar will only be refreshed when
  you actually do something.

- reworked internal Startup.DM a bit

2.2b4 (97-03-24) 58716 Bytes

- removed more dead code I found reading the source in my breakfast-break

2.2b3 (97-03-23) 58772 Bytes

- removed all 1.3 compatibility routines
- removed "ShowPic" and "Backpattern" - saved 6.6k
- included VersionString
- general cleanup of the source - removed all dead sections
  239408 Bytes -> 164451 Bytes total

2.2b2 (97-03-20) 65632 Bytes

- modified "About" to pop up an easyrequest() requester, this saved 3.5k and
  will do nothing on OS below 2.0
- Option "Fast" added to SortBy for testing purposes
- removed "Play" for being a buggy, obsolete, assembler part of the code,
  saved 5.5k
- removed "Format" and "DiskCopy" because I think that
  a) nobody is really using a disk-only system anymore
  b) it would have been too much work to update them
  c) it was nice to save again 5.3k
```

And please be patient, this not a weekend-project!
I printed the source - 130 pages ! - now my daily work is to drive to the company I‘m working for, xerox a chapter or two from the source, reading it during my breakfast and lunch-breaks, marking some things.
After working 8-10 hours for my company I drive my bike home.

I have the source for ten (10!) days now!
This will take some time, but I‘m not giving it up!!

- 97-03-11 recompiled using SAS-C 6.57 and Includes 40.15

From the file "Programme_Changes" that came with the cover disk:

```
v2.1c Fixes:
  Printer setup now shows the proper values instead of always 0.
  This must be what’s causing the other problems with output, since
  you can’t see what the values really are.

  Print doesn’t switch to HEX when the file is less than 199 bytes.
  PrintDir now works.
  PrintDir will exclude escape code if the output is not PRT:.
```

I’ve also added *some* AGA support.
I had a 4000/Toaster to test for a couple of days.
All but HAM8 hires work fine.

DISKMASTER 2.1 -> 2.1a CHANGES:

```
· Fixed S<->D swap bug; now no longer crashes machine.
· Fixed the addition of Hex information when saving a STARTUP.DM file;
  this bug especially noticable on floppy-based systems.
·~Fixed Progressive logo.
· Fixed CHKsize bug; When a SOURCE device was checked, if there wasn’t a
  DESTINATION chosen, the program would bomb.
```

DISKMASTER 2.04 -> 2.1 CHANGES:

```
· Added FFS to format.
· Added HD formatting, but not tested.
·~Added Backdrop pattern/picture for custom screens.
·~Diskchange and directory modification detect enhanced.
·~2.0 pens are saved in the config.
·~Fixed crash when deleting files when sorting by date.
·~Project icon DefaultTool path is expanded so it will always find DM.
·~Fixed font selection problem in printer setup.
```

1.101 What`s next?

```
------------------------------------------------------------------
                 The road-map for future-releases:
------------------------------------------------------------------
```

- Getting rid of all this Kick 1.3 hardcore intuition code!!!

- Fix all the bugs.

- Include all your wishes.

- A "Read %s HEAD" Option

- Gadtools GUI

- More keys used and accessable, e.g. cursor-controled windows, CTRL,ALT for key-commands.

- Directory buffers

- Progress Indicator

Most certainly not in this succession.

There will be no version for Phase5‘s PPC boards since I don‘t think that they are of any use when a complete PPC System is available for nearly the same prize and it looks like we have a ported AmigaOS soon.

And besides this DM2 is not that CPU intensive and with optimized algorithms it will also fly on your system!

Future changes to the guide:

- Add much more information to the list and provide more "exotic" examples
- More scripts to be bundeled with this release

- A survey on the users demands for new functions in DM2.

- A tutorial on using DM2.

1.102 Discuss your problems with friends...

```
------------------------------------------------------------------------------
                            DM2 Mailinglists:
------------------------------------------------------------------------------
```

There are two lists installed since 97-06-09 21:45!

And now you may subscribe to the first: dm2-forum

Just send an e-mail to: majordomo@on-luebeck.de
And include this line in the body: subscribe dm2-forum
Additionally you should include this line: help

This is the list for discussion about all topics you have concerning DM2.

After receiving the "welcome to..." from majordomo you are using the list by sending mail to: dm2-forum@on-luebeck.de
As a kind of return-receipt, your mail will be "bounced".

You should filter out the mails from the list by scanning for:

Sender: owner-dm2-forum@on-luebeck.de

Thanks to Hannu Nevalainen for correcting me with this.

The second list - dm2-beta - is not yet ready for subscription.

1.103 Howdy Friends!

```
------------------------------------------------------------------------------
                                      Credits
------------------------------------------------------------------------------

Special Greetings to:

Greg Cunningham    - Thank you!
Jerry W. Rubemeyer - You finally ended my quest - many thanks!

The following persons gave me a hand on creation of this document:

Mike Hendren - thank you for providing the DiskMasterII.Doc!!
J.M.S.oftware - whoever you are, thanks for typing the manual!
Jody Tierney - thanks for reporting the bugs in DM2.guide 0.13 and in <=0.39!
Bryan K. Williams - thanks for reporting the bugs in DM2.guide 0.15!
Manuel Martin-Vivaldi - thanks for reporting the bug in "Find" ‘till 0.22!
                        Thank you also for suggesting the "Question & Answers"
                        and for sending some text for the
                        tools section, describing HTMLess.

Fast Greetings to:

Martin Berndt - Thank you for MultiCX, it‘s still my favourite!
Nico Francois - Wherever you are...
Dave Haynie   - Thank you for the hardware!
Heinz Wrobel  - Thank you for updating the OS!
The OS3 Team  - Very nice work guys, but please complete it someday...

Personal Greetings to:

Richard Backhaus - my all-purpose coding-slave :-)
Jacques Bojart   - long term brother in arms...
Henrik Doerfler  - I still remember the day when I came from Volker‘s place...
Nicolas Franck   - I really want that ProWizard update...
Thomas Kelly     - Thank you for a very interesting Mail-Thread!
Magnus Kutz      - What about a trip to the "Body‘n‘Soul"?
Ingmar Mix       - I won‘t trust you again in dating with girls...
Martin Sass      - Let‘s have a party again!
Patrick Schimpke - The szene‘s dead anyway... :-(
Timo Treeter     - Thank you for the party-panic, honey.. :^)

And don‘t feel angry if you think I should have mentioned you.
Just drop me a mail and we‘ll see... 8-)
```