

NextE

COLLABORATORS

	<i>TITLE :</i> NextE		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 31, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	NextE	1
1.1	main	1
1.2	general remarks	2
1.3	start	3
1.4	configuration	5
1.5	move	6
1.6	messages	7
1.7	search	7
1.8	search by mouse	9
1.9	search in binaries	9
1.10	blocks	10
1.11	count	10
1.12	statistics	10
1.13	print	11
1.14	print with page numbers	12
1.15	print modes	12
1.16	initializing printer	13
1.17	window sizing	13
1.18	load	14
1.19	line highlighting	15
1.20	file conversion	16
1.21	html	17
1.22	amigaguide	18
1.23	pictures	19
1.24	info	20
1.25	file deletion	20
1.26	%-display	20
1.27	archives	21
1.28	tabs	22
1.29	megafiles	23

1.30 settings	23
1.31 editing current file	23
1.32 copy files	24
1.33 font selection	24
1.34 reformat	24
1.35 changes	25
1.36 thanks	26

Chapter 1

NextE

1.1 main

NEXT 3.01

(c) 1993-2000 Jürgen Klawitter

FREWARE

NEXT needs OS2+

NEXT is mainly a text reader but can also be used as (offline) browser, Amigaguide replacement, datatypes viewer, file converter, print utility, dearchiver and more ...

INSTALLATION & SHORT INSTRUCTION

Copy NEXT to C: or somewhere else in your path. Reqtools.library, (c) by Nico Francois, should be in LIBS:, although NEXT can work without it. If you use NEXT frequently you should insert a line like "resident c:next pure" in your user-startup.

Further on amigaguide.library is needed if AmigaGuide files are to be displayed as hypertext. XPK-Libraries, LhA, LZX, UnZip, UnRAR and GZip are required for xpk-compressed and archived files respectively.

You can use NEXT without reading this document although you probably will not detect all features. Most commands are compatible to those of More, MuchMore or PPMore. Press "Help" in case of need. The Help window can be used as popup menu: Doubleclick on a line or use cursor keys + Return to start an action.

Contents

General remarks
Start, options

Highlighting a line
File conversion

Configuration	HTML
Moving through text	AmigaGuide, datatypes
Messages and input	Viewing pictures
Search	Info, filenote
Search by mouseclick	Delete current file
Searching text in binary files	%-Display
Marking a block	Archives
String count	Tabs
Word statistics	Megafiles
Print/Save	Settings
Print with page numbers	Editing the current file
Print modes	Copy files
Initializing printer	Font selection
Changing window size & position	Reformat current text
Loading a new file	Changes in V3.01
Thanks, Author	

1.2 general remarks

General remarks

NEXT will use the system font if at least 79 rows of text can be displayed with this font. Otherwise topaz 8 will be used. If the default font is smaller than topaz 8, 80 rows will be displayed. You can override this behaviour by options FONT and ROWS (see below).

NEXT formats the text before displaying it. That means, lines are wrapped at word boundaries if they are too long and nonprintable chars are converted to spaces. ESC-sequences for color, style and printing will not be touched. Formfeeds are left intact, too, but will be displayed as L. Formatting is done very fast, but will take some instances if a file is very long.

Formatting speeds up text display, also binaries can be read like ordinary text files. The disadvantage is that the length of lines stays constant if window width or tab size are changed. You have to reload the current file in these cases to adjust lines again.

The window title contains some useful informations, two of which have to be explained. Example:

```
2/16 AMINET11:Aminet/AMINET (262.941 bytes)          DV*   50%
^                ^
1                2
```

- (1) If a file couldn't be loaded completely into memory, you can see here which part of it is currently displayed (part 2 of 16 in this case).
- (2) At this position some status informations may appear:
 - D = currently displayed file has been deleted
 - V = commands "VIEW archive" and "UNPACK archive" can be used
 - * = a pattern exists which can be used to scan a directory, see Load next

Some terms used in this doc should be explained to avoid confusion:
current file = file which is currently displayed in NEXT's window

current directory = directory of current file (not that of the shell)

1.3 start

Start

Workbench:

=====

1. Click on NEXT icon and doubleclick on the icon of a file or directory (holding down the shift-key)
2. Set NEXT as the default tool in a text icon and doubleclick on that.

If you only doubleclick on the NEXT icon, a filerequester will appear.

Shell: > next [file|directory] [options, see below]

=====

All arguments are optional and can be typed in any order. Without arguments the filerequester will open. The same is true if you provide the name of a directory. If a wrong name has been given, NEXT tries to extract the parent directory from the pathname and presents that.

Instead of a file or directory a pattern with AmigaDos wildcards can be used. Patterns for directories must be terminated by a "/".

After the first file has been loaded, you can continue to scan the directory for other files which match the provided pattern, using the command "LOAD next".

OPTIONS

=====

There are the same options for Workbench, Shell and config file. When starting from Workbench they can be defined as tooltypes in the project icon. Tooltypes in NEXT.info are ignored. Use ENV:next.prefs instead to define basic settings.

AGLIB (formerly @LIB)

AmigaGuide files are displayed using amigaguide.library.

PURE (formerly @PUR)

The default behaviour of NEXT is to convert AmigaGuides, HTMLs, MSWord files etc. to ASCII/ANSI text. With option PURE all files are displayed as they are.

DTMode

Switches the datatype mode on at program start. Files, e.g. pictures, are then displayed by the appropriate datatype. OS 3+ is required. You can turn this mode on after start by pressing Shift-A.

ROWS(=R) n

This option sets the maximal number of text rows (columns) to be displayed on screen (default 79). This will also be the maximal print width. Valid values are 40-126. Smaller values will be set to 40, larger ones to 126.

LINES(=L) n

Sets the number of text lines to be displayed at start, is ignored later on. Smallest value accepted is 8.

WX, WY n

x- and y-positions (in pixels) at which window will open.

CENTER

Centers the window at start. This can only be done if you have defined window dimensions by ROWS and/or LINES which are smaller than the screen.

RIGHT

Places the scrollgadget into the right border of the window (default is lower border). This reduces the text width to 77 chars on a 680 pixel screen with topaz 8 but is acceptable if you have a higher screen resolution (e.g. gfx board).

FONT(=F) name/n

"name" is the fontname WITHOUT suffix ".font", n is the size of the font. Example: FONT Courier/13 or F Courier/13
The font must be of fixed width, otherwise it will not be accepted. Fontnames should be written case sensitive, especially if you want to use topaz/8 or topaz/9.

TABS n

Defines the tab width (n=2-16). Default 8.

LHA <path>, LZX <path>, UNZIP <path>, RAR <path>, GZIP <path>

Defines where an archiver will be looked for in case an archive has to be extracted or viewed. Example: lha c:lha.

Paths which contain spaces must be enclosed by ". You should define the paths in ENV:next.prefs otherwise you will be asked when an archive occurs.

RES

Makes unpackers LHA, LZX und UNZIP resident when needed the first time (GZIP and UnRAR are not residentable).

This increases speed of archive operations and decreases number of disk accesses. All unpackers are removed from resident list (and memory) when leaving NEXT.

ED <path>

Path of the editor to be called if you want to edit the current file.
A path containing spaces must be enclosed in "".

TEMPDIR <path>

Defines the directory where the archivers will write their temporary files to. The directory must exist. It should be in RAM if you have enough memory. Default is T:. Within TEMPDIR a subdirectory "x" will be created if it doesn't exist.

NOFLUSH

After quitting the program all temporary files including subdirectory "x" will be deleted by default. You can prevent this by choosing NOFLUSH.

TOPMARGIN <n>

Inserts as many linefeeds as specified on top of a printed page. This may be useful if your printer doesn't understand the printer command "top margin" like my Canon BJC-4400.

ASK

If defined a requester will appear every time before loading a file by "LOAD next". The advantage is that you can see which file will be the next one (normally you don't know this) and you can decide between Load, Skip and Cancel.

Options are treated in the following order:

1. Shell or Workbench options > next.prefs > internal defaults
2. FONT > ROWS,LINES > WX,WY > defaults

That means: A given font will be used, even if a given number of rows can't be realized. ROWS has priority above WX and default rows. If the number of given rows is too big, NEXT tries to show as many rows as possible. topaz/8 will be used if the default font is bigger.

The following table gives some examples of usable fonts and the maximal number of rows on a standard screen with 640 pixels.

fontwidth rows font/size

6	105	thin	topaz/8	Courier/11			
7	90		Courier/13				
8	79	topaz/8	topaz/11	pearl/8	look/8	SanS/11	
9	70		Courier/15				
10	63		topaz/9				

1.4 configuration

Configuration

When started NEXT looks for two files:

```
ENV:Next.prefs
ENV:Next.files
```

Next.prefs contains information about configuration. Next.files may contain a list of frequently accessed files and directories which can easily be loaded after start of program by pressing "+" (see MyFiles).

Both files can be written with an editor and must be saved to ENVARC: and ENV:.

Each entry must be placed on a separate line. Comments must start with a semicolon. Paths containing spaces have to be set in "".

```
-----
;example for next.prefs
```

```
font=Courier/13
rows=90
lha=lha      ;no path if LhA is resident
lzx="work:"my archivers/lzx"    ;path contains a space
ed=c:dme
res         ;make unarchivers resident
;end of config file
-----
```

The first line in next.files has a special meaning:

If NEXT is started without a file argument it takes the first name within Next.files and loads it. If you want NEXT to behave as usual (opening a filerequester) you should write "\" on the first line.

```
-----
;example for next.files
```

```
\      ;show current directory on start
work:pd/DiskSalve/Locale/docs/english/disk salv.doc
\      ;actual directory, for better arrangement
cd0:aminet/index    ;for AMINET CD's
cd0:aminet/pix
\
"work:my pictures/#?"      ;wildcards are allowed
\#?.readme      ;scan current directory for .readme's
\#?      ;scan all files in current directory
docs/log font thinpaz/8    ;file and option mixed
;EOF
-----
```

1.5 move

Moving through text

Main window

As mentioned above key commands for moving are like those of other common textviewers. Space goes down one page, Backspace up one page. With cursor keys the text can be scrolled up or down. Keys of numeric keypad can be used, too. Text can be also moved by the scroll gadget at the bottom border of the window. Press Help in case of need.

Goto (G) jumps to the text position you want. You have to provide a percent value if the status field in the window bar displays percent otherwise a line number.

Jump (J) jumps to the position where you jumped from with one of the commands Top, Bottom, Search or Jump to mark. If you are e.g. somewhere in the text and you jump to the bottom you'll get back to the previous position by pressing "J". Jump back stores the position every time before a jump is performed. A jump must be longer than one page.

List windows

The commands RELOAD, LOAD MyFiles and VIEW archive display file lists in their own window. You can move through these lists with the same keys as in the main window (cursor keys, BS and Space). A file is selected by double click or by moving through a list with cursor up/down and pressing Return afterwards.

You can search through the lists by pressing "S" or "." and entering a string or pattern. "N" continues search, "P" searches back.

1.6 messages

Messages and input

Error messages and informations are shown by requesters which can be closed in most cases by pressing any key. If there two or more gadgets Return ist equivalent to a click on the leftmost gadget (OK, Continue..).

Several commands afford an input (Search, Load, Print..) which is done into an input window with a string gadget. Empty input (nothing typed and Return pressed) aborts the respective command.

Apart from printable chars following control chars are accepted if the filter for string gadgets in Prefs/IControl is disabled:

TAB	Tab key
ESC	Esc
CSI	Alt-Esc
Linefeed(LF)	Ctrl-J
Formfeed(FF)	Ctrl-L
Return(CR)	Ctrl-M

1.7 search

Search (Search ci/cs, Next, Previous)

Strings can be searched case sensitive (cs) or case insensitive (ci). They can contain control chars like LF. This enables e.g. searching of strings which continue on next line or finding only words at beginning of a line.

NEXT marks all occurrences with background color 2. Control chars in a search string are converted to visible chars.

In contrast to most text readers NEXT marks ALL occurrences on screen, not only the first match. This is faster and gives much better survey.

Search starts from current position (displayed text included) if a new string has been given or if search direction changes.

Searching forward the first match will be displayed 2 lines below top of window (if possible). Backsearch displays it 2 lines from bottom. So you can see what is above/below.

Usually search goes downward after string input. But you can force immediate backsearch by clicking on the gadget "Search back" or by using the shortcut Right Amiga-B.

Pattern search

=====

From V1.8 on strings to be searched may contain wildcards which are equivalent to the AmigaDos wildcard "?". They replace any alphanumerical character. Input of wildcards is done by Help key.

EXAMPLE: Input is "b?g". As result bug, big, bog, bag... will be found.

If a pattern contains nothing but wildcards NEXT will look only for words with the same length as the pattern. (It wouldn't make much sense to display all occurrences without this restriction.)

In order to search for words with length n the input can be "[n]".

Example: [15] starts search for words which contain 15 chars.

=====

The special search abilities of NEXT may give nice effects. The following demonstrations will only work if this text is displayed by NEXT (without amigaguide.library).

EXAMPLES:

1. Please press following keys in the same order: S, TAB, Return.

;

Result: All TABs are marked (see line above).

2. Pattern search: Press S, type "th??" and Return. Input of "??" by Help key.

that they others whether through the pathname either

3. Press S, Alt-Space, Return.

```

~~~~~
~*\textdegree{;}~\textdegree{;}*~ ~~
~/=\ ~ ~~~
~~~~~
~~ ~ ~ ~
~ ~~~~ ~ ~ ~~~~~ ~ ~ ~
~~~ ~~~~~~ ~~~~~ ~ ~ ~ ~
~~~~ ~~~

```

This image is composed of spaces with Code 160.

1.8 search by mouse

Search by mouseclick (MSearch)

Occurrences of any word on screen can be simply searched by holding down Ctrl and clicking on the first (visible) char. This is a nice feature not offered by other text readers. Search is ci if you press Ctrl alone but will be cs if another qualifier is pressed additionally.

Direction of search depends on the mousebutton pressed:

```
Ctrl-LMB (left mousebutton) -> forward search
Ctrl-RMB (right mousebutton) -> backward search
```

The selected word will be inverted as long as buttons are held down. Except the first char all others must be alphanumeric. Point, underscore and hyphen are treated as chars if they are placed within the word.

EXAMPLES:

clicked	marked/searched
Workbench:	Workbench
A600/1200	A600/1200
;comment	;comment
; comment	; comment
Nölle-Neumann	Nölle-Neumann
12.5.94	12.5.94

1.9 search in binaries

Searching text in binary files (XSearch)

This function finds any readable ASCII text in programs and other binary files. It may be useful e.g. if you want to know which libraries, devices or files a program needs to work.

Strings are considered as ASCII if they consist of at least 4 alphanumeric chars. Change of upper case to lower case may occur only at beginning of a string, otherwise it will not be marked. There are of course many char combinations within a program which are senseless, but hitrate of this function is quite good.

XSEARCH starts at current position and can be continued by pressing "X". Backsearch is not possible.

If Shift-X is pressed words consisting of 3 chars only can be detected. Of course, much more "garbage" is found in this mode.

1.10 blocks

Marking a block (Mark)

A line can be marked with Shift-RMB or Shift-LMB. The positions are stored and indicated by special signs at start of line (1 for LMB, 2 for RMB). Marks can be shifted by simply Shift-clicking at another line.

Marks can be removed by pressing Del or by Shift-clicking on the same line a second time.

Two marks define the beginning and the end of a block, marked lines included. The order of marks (1/2 or 2/1) is unimportant. If only one line has been marked this line will be taken as a block.

The following operations can be applied to a block:

- Print
- Save
- Word statistics
- Jump to mark 1/2
- Count block

1.11 count

String count (Count ci/cs, Count old)

This function counts all occurrences of a string or a single char within current file. With "C" count is case insensitive, with Shift-C case sensitive. The result of count is displayed in a requester. The title of the requester window shows whether count was ci or cs.

Count can be restricted to an existing block. Press the "Count block" gadget in this case after input.

A counted string can be searched later on with "N" (NEXT) or "P" (PREVIOUS). On the other hand a previously searched string can be counted by pressing Amiga-C or Amiga-Shift-C (block). The mode (cs/ci) will be the same. If the counted string is a pattern consisting of ??? only the number of words having the same length will be reported, e.g. "word[3] found 34 times". [3] indicates that length was 3 chars.

The count function is mainly for analysing texts. You can also use it to check whether a certain string occurs in a file without leaving current text position.

1.12 statistics

Word statistics (Count words)

This command is started by "W". It counts the words of the whole text or within a block (if defined). Words which are separated by a hyphen at line end are considered as ONE word. The result is presented in a window:

The first line shows the total number of words and average word length. The table below presents the occurring lengths, number of words (n) with same length and portion of words with same length (%) in text. Maximal word length is 30. Longer words - they occur very rarely - will be added to length 30.

"Words" are defined here as strings of alphanumeric chars which are separated from one another by non-alphanumeric chars except point, hyphen and underscore.

```
EXAMPLES:  print/save    2 words
           Next.doc      1 word
           A-Z           1 word
           A - Z         2 words
           4711          1 word
```

Word statistics are senseless for binary files. In these cases the contents of ASCII strings in % is displayed.

1.13 print

Print/Save (Print)

This function saves the text or part of it to PRT: or to a file. Pressing Shift-P will save the whole file, pressing "*" will save the currently marked block.

NEXT checks if printer is online (for parallel port only). If printer is online PRT: will be the default output, else you have to enter the filename you want to save to. "PRT:" can be replaced by a filename if you don't want to print.

A block can also be saved to clipboard (Unit 0). The input window has a gadget "Snip". Click on it or use the shortcut Amiga-S for a clipboard copy.

If the file you want to save to already exists, you'll be asked whether it should be overwritten or whether text should be appended. Press "O" to overwrite or "A" to append. Any other key will abort. The append mode may be useful if you only want to collect some important parts of a text file.

If output goes to PRT: the number of pages and sheets needed are calculated and shown in a requester. You may start printing now by pressing Return or abort with any other key.

While printing NEXT uses the page length defined in printer settings and sends a formfeed to the printer after each page (except last page). This is NOT the case if the text already contains formfeeds or if output goes to a file.

1.14 print with page numbers

Print with page numbers (Print pagenum)

If you want to print/save with page numbers and a heading on each page press Amiga-P (whole file) or Amiga-* (block). You'll be asked for the text of the heading. Default is the filename, which can be taken over by Return. After that you'll be asked for page length, which must be at least 3 lines. Default is the page length defined in printer settings.

After page length has been entered NEXT calculates the number of sheets needed and informs you by a requester. You can now start the print or cancel.

The heading looks like this by default:

```
Next.dok~1
```

Here comes the first line of text...

Modifying page header

=====

1. Empty input for title text will cause NEXT to print without a title line but with the page length you have entered. So you can use other page lengths than default length.
2. If you only type one or more spaces as title text, title header will be filled with spaces but page numbers will be printed.
3. Style and position of header text can be influenced by a number of options which can be switched on or off by clicking in the settings window. Press "" to open the settings window.

BOLD, UNDERLINED and ITALIC can be chosen for text style. They can be combined. Default is UNDERLINED.

The new option CENTERTITLE centers the title (surprise). If the title consists of spaces only the pagenumbers will be centered.

1.15 print modes

14.Print modes

In the settings window (opened by "") four different print modes can be selected by a click (or unselected by a second click):

```
SMARTPRINT
BOOKMODE
ODDSFIRST
PAUSEPRINT
```

- (1) SMARTPRINT tries to be intelligent and varies the pagelength a bit according to structure of text. That means:

If at pageend only one line of a paragraph is left that normally would be printed on next page, it will be printed on the same page.

If at end of page the first lines of a new paragraph or a heading appear, they will be printed on the next page. Thus, a printed page may be 1 line (max.) longer than defined by printer prefs or input. It may also be several lines shorter.

The `smart_print` mode is `DEFAULT`, because it gives the better results in most cases. If `smart_print` is switched off, all pages will be printed with constant page length. Minimal page length for `smart_print` mode is 11 lines. For shorter pages the normal print mode will be used.

HINT: Print a file with `pagetitle` and page numbers to `ram:test` first, load `ram:test` and look at the result. Then print it to `PRT:` if all is OK. Use `Shift-P` (NOT `Amiga-P`) this time.

- (2) `BOOKMODE` can be used if a text is printed with page numbers or page heading. If this mode is on the heading of pages with even number will be printed in reverse order. Page numbers will appear on the left side and the title string on the right side. Thus, page numbers will always be at the outer margin like in books.
- (3) If `ODDSFIRST` is on all pages with odd number will be printed first, after that the pages with even number will follow. This may be useful if you use a printer which loads single sheets automatically. After odd pages have been sent to printer a requester will appear. Wait until printer has stopped, then turn the stack of sheets and feed them back into the printer. Press the gadget to start printing of even pages.
- (4) `PAUSEPRINT` stops every time a page has been sent to printer and opens a requester. Printing will continue if you press any key or the "Print"-gadget. To abort printing you have to press "C" or the Cancel-Gadget.

1.16 initializing printer

Initializing printer (Init printer)

After pressing "=" `NEXT` will try to initialize your printer according to printer settings. Printer must be online.

This function may be useful if printer was switched off for some time or after aborting a print.

1.17 window sizing

Changing window size and position (Size/Move window)

Pressing `F1/F2` keys changes size of window. 3 sizes are available: maximal, minimal and middle. `F1` toggles between min. and max. size, `F2` between middle and max..

By default middle size is half as high as maximal, but can easily be changed by dragging the size gadget. `NEXT` remembers the new size und

will use it on next occasion.

MOVE window (F3) moves the window down if it was in the upper half of screen or up if it was in the lower half. If the window is as big as the screen nothing will happen.

1.18 load

Loading a new file (Load/ReLoad)

NEXT offers several variants to load a new file:

1. Load req(uester) (L)

Opens the filerequester and presents the directory of current file.

2. Load man(ually) (Shift-L)

The complete path of the directory or file wanted has to be entered into the input window. If the path contains spaces, you must place it in "".

The current directory may be abbreviated by "\".

You can also give a pattern instead of the filename and use the same options as in shell commandline. E.g. if input was "font topaz/l1" this font will be opened and the current file will be reloaded.

3. Load newwin (Amiga-L)

Input as described for Load man(ually). NEXT will create a new process which opens its own window or presents a filerequester if a directory has been entered. Empty inputs are ignored.

4. ReLoad (<)

Enables you to load a file again which already had been viewed during a session. NEXT saves the names of previously loaded files into a buffer of 800 bytes. If size of buffer is not sufficient older names are replaced by newer ones.

After pressing "<" a small window which presents all names saved will open. You can select a name by doubleclicking or by moving through the list of names with cursor keys (Up/Down) and finally pressing Return. If nothing has been selected Return aborts this function.

ESC, Space and closegadget can be used, too.

5. Load next (>)

What happens when you press ">" depends on what you have done before.

If you have previously loaded a file by using a pattern (e.g.

work:source/#?.s) NEXT will scan the same directory (work:source) and look for the next file matching the pattern (#?.s). An existing pattern is indicated by a "*" in the status field (window title bar).

If no pattern exists the current directory will be scanned using the pattern "#?.<suff>" where <suff> is the suffix of the current file.

If there is no suffix the pattern will be "#?".

By repeatedly pressing ">" you can view all files in the scan directory matching the current pattern. .info-files will be skipped. Which file

will be loaded next depends on how the files are arranged on disk. It's NOT in alphabetical order! I recommend to use the Option ASK if you wish to decide whether the next file is to be loaded or not.

If you press ">" in combination with Alt or Amiga keys the current directory will become the new scan directory and a new pattern - depending on the suffix of the current file - will be defined, e.g. "#?.readme" if the suffix is ".readme". This is especially useful when looking into directories on AMINET CDs: Load a .readme file, press Alt+> and the next .readme will be loaded.

6. MyFiles (+)

Opens a window which shows the files, directories and options written by you in ENV:next.files. They are selected as described under RELOAD.

7. Load by appwindow

Drop an icon or a filename within a workbench window on NEXT's window and the corresponding file will be loaded. If the icon belongs to a directory the filerequester will open.

8. Back/Forward (Shift-Left/Shift-Right)

Steps back or forward in the file history and loads the previous/next file. If the end of history buffer is reached (contains 5 entries) the file at the other end of the buffer will be loaded. In other words: If you have stepped back through the last 5 files (5->4 4->3 3->2 2->1) the next step back loads file 5.

9. MatchFiles (M/Shift-M)

After pressing Shift-M you have to enter a directory and a pattern separated by at least one space. NEXT will now search for files matching the pattern in this directory and all subdirectories.

Examples: cd0:aminet/gfx #?.readme
"Ram Disk:" #?.prefs ;use "" if names contains spaces!

During search the number of currently matched files is shown in the title bar. Search can be stopped by Esc or Space.
The files found are presented in a list window and can be loaded as described above (ReLoad, MyFiles).

If you want to load another file press "M" to open the list window again.

1.19 line highlighting

Highlighting a line (Show line)

After pressing Alt-MB (left or right) or Amiga-Alt the line at mouse position will be inverted. This makes reading of tables etc. easier. The line number is displayed in the titlebar.

1.20 file conversion

File conversion

The following file types are automatically converted to readable text if PURE mode is not turned on:

HTML, GUIDE, (Mac)WORD, WRITE, WPD, (Mac)RTF, HLP, MAN, FTXT, FW, WW, DosTEXT, MacDOC, MacBIN and MacTEXT.

Switch conv(ersion) (key "!") loads the current file again and uses the opposite conversion mode. A DosText will be converted to ANSI and vice versa. This applies also to MacTEXT, GUIDE, HTML, WORD, WRITE and RTF.

So you can see very easily what's e.g. 'behind' a converted HTML file by pressing "!" and switch to converted mode back by another key stroke.

WORD/WRITE

NEXT extracts the pure text from files produced by the Microsoft programs Word and Write (suffixes .doc and .wri). Footnotes, endnotes and annotations within WORD documents get continuous numbers which correspond to references in the text:

```
footnotes  [1], [2], ...
endnotes   [E1], [E2], ...
annotations [A1], [A2], ...
```

WPD (WordPerfect Document)

Documents of WordPerfect V5-V7 will be converted. The result may be not very perfect due to lack of documentation and test files. Unknown chars are converted to "\textdegree{}".

RTF

This is the Rich Text Format which can be produced by MSWord. NEXT extracts the text and converts style instructions (bold, italic, underlined). Footnotes will be numbered and moved to the end of text.

HLP

Documentation of programs running under Windows ist often stored in special help files with suffix .hlp. NEXT extracts the text within these files. Unfortunately about 2/3 of HLP-files are packed so that conversion is impossible.

MacTEXT, MacBIN, MacDOC

These are text formats created on Macintosh computers. Special chars are translated to ANSI chars and CR to LF.

FTXT, FW, WW

FTXT: This IFF format is used to save text to the clipboard.
FW: Documents produced by Softwoods FinalWriter/FinalCopy.
WW: WordWorth document

MAN

Unix and Linux "man pages" containing formatting instructions for nroff are converted to readable text. Results are satisfying in most cases but can be bad if the file contains macros or unknown instructions.

DosTEXT

Texts using the MSDos charset (codes \$80 to \$9F) are converted to ANSI (=Amiga) codes. It's not easy to detect which charset has been used. So errors may occur. Use the Switch conv(ersion) command in this case.

1.21 html

HTML

Web pages are written in HTML (HyperText Markup Language). These files can be recognized by their suffix .htm or .html. They contain a lot of instructions for WWW-browsers (so called tags and entities) which make HTML files almost unreadable. NEXT converts and formats HTML files by default. Conversion does not take place if PURE mode is on.

Some hints:

- * All tags and entities of HTML 3.2 are converted if possible.
- * Some HTML file contain very little or no text at all. So, don't be afraid if you see an empty window after conversion.
- * Text is formatted according to window width. Make the window as wide as possible before loading a HTML file.
- * The title of a Web page is displayed in [] in the upper left corner.
- * Conversion of tables is nearly impossible and therefore done in a very simple manner.
- * Headings (h1...h6) are indented and followed by a blank line. Main headings (h1,h2) are display in colour 2, h1 additionally in bold style.
- * Frames can't be displayed but references within frames are shown, e.g. "FRAME <framename".
- * References are marked by color 3. When you click on them the target is shown in the title bar as long as LMB is held down. After releasing LMB the target will be shown. You can abort the action by drawing LMB above or below the line where the reference is. LMB must be still held down.

Following types of targets are possible:

```
file.html      target is another file which will be loaded
file.html#mark target is a special position within another file
#mark         target is another position in the current file
```

Sometimes references are represented by a picture and not by a string. In these cases NEXT shows the symbol [] which can be clicked, too.

HTML files very often contain another type of pictures (inline pictures) which appear in "real" browsers as buttons or something like that. These

are represented by the symbol [⌘]. If you have OS3+ and "DTMode" (see below) is active you can view the pictures by clicking on that symbol.

Searching references (Browse down/up)

By pressing "#" you can jump to the references in HTML files and Links in AmigaGuide files respectively. Every reference found is displayed with background color 2 and the target is shown simultaneously in the title bar. You can jump to the target by pressing Return. Pressing "#" again will show the next references downwards. Using Amiga or Alt in combination with "#" inverts the direction of search.

Switching references off

Display of references and picture symbols can be switched off by inactivating the option MAKELINKS in the settings window. This may be useful for printing the current document. It also speeds up the conversion of HTML and AmigaGuide files a little bit and saves some memory.

1.22 amigaguide

AmigaGuide, datatypes (AGDT once, DTMode)

There are 3 display modes for AmigaGuide files:

1. converted (default)
2. by amigaguide.library (AGLIB mode)
3. without conversion (PURE mode)

Modes 2 and 3 can be turned on by options AGLIB and PURE at programm start or can be selected later on in the settings window.

The AGLIB mode uses the amigaguide.library (like Multiview or AmigaGuide). If the amigaguide.library can't be used for some reason (low memory, no library present) files will be converted.

The PURE-mode shows AmigaGuide files with all the special instructions.

By default all AmigaGuide files are translated to readable text. This means all instructions (@NODE etc.) are stripped off and replaced by equivalent escape sequences. The end of a NODE is marked by a "#" in the converted file. LINKS will be rendered in color 3. Clicking on a link will cause a jump to the referenced position or file. As long as the left mouse button is held down the reference is shown in the window title. Drawing the pointer one line up or down while the mouse button is still pressed will abort the action.

Advantages of conversion:

In contrast to the presentation by amigaguide.library the guides are not fragmented into small pieces and you can use the search and print functions of NEXT. The converted file is shorter than the original and can be saved, of course.

AGDT once (=AmigaGuide & DataType)

If the current file is an AmigaGuide file (PURE or converted) and you want it to be displayed by `amigaguide.library`, press "A". The library opens its window in front of the NEXT window.

Starting with OS3.0 not only AmigaGuide files can be viewed with this function but every filetype which is represented by datatype (binary datatypes excluded). Since `amigaguide.library` V39+ uses datatypes itself NEXT simply passes the file to the library. In case of pictures NEXT uses its own routine (see following chapter).

DTMode (=datatype mode)

Shift-A switches this mode on or off. If switched on the mouse pointer changes and all files loaded afterwards (except guides and binaries) will be displayed by datatypes. OS3+ is needed.

1.23 pictures

Viewing pictures

All types of pictures including icons can be displayed if the corresponding datatypes are available. There are different ways to display a picture:

1. Load a picture (DTMode is off) and press "A".
Disadvantage: The file must be loaded by NEXT and the datatype as well. This consumes time and memory.
2. Switch DTMode on (Shift-A) and select a picture file by one of the LOAD functions. The picture will be immediately displayed. The previously loaded text file stays in memory and can be viewed again after the picture has been left.
3. If you want to see all pictures within a directory switch DTMode on and use a pattern, e.g. `work:pics/#?.iff` or `work:pics/#?`. By pressing ">" repeatedly you can now view all pictures matching the pattern.

A picture can be quitted by pressing any non-qualifier key.

Info about the picture

You can get some information about a currently displayed picture by pressing "I". A requester then shows datatype, width, height, depth, number of colors and screenmode. The colors of the requester depend on the palette of the picture, by the way.

Screendump

You can print the currently displayed picture by clicking on the Print-gadget of the requester. The print is done according to `PrinterGFX` settings.

If you are owner of a gfx card the print result will be garbage in most cases because the `dumpscreen` function of `printer.device` works only with standard (Amiga) screen modes.

You may use a tool like Olaf Barthel's `P96_PrinterPatch`, which must be

started before printer.device is opened the first time.

1.24 info

Info, filenote (Info/Filenote)

Info

Pressing "I" pops up a requester with following informations:

- date of creation/last change of file
 - protection bits
 - file type (GUIDE, HTML, (Mac)WORD, WRITE, (Mac)RTF, DosTEXT, FTXT, FW, WW, (Mac)TEXT, MacBIN, MacDOC, BIN)
- For WORD and WPD the versions are shown, too.
- information if crunched or decoded and type of cruncher (PPData, XPK compressor, SZDD, HQX). SZDD is the identification string for files compressed with MSCompress.
 - number of bytes removed by text conversion. This is mainly debug information for me. Small numbers are due to stripping off carriage returns and trailing spaces.
 - :filenote
 - number of lines and longest line ("rows")
 - number of lines and size of block if marked
- ~~~~~

After a separating line scan directory and pattern used by Load next are shown if there are any.

Filenote

If you press Shift-I a filenote can be entered for the current file. You must put it in "" if it contains spaces. A previous comment can be deleted by typing "". Simply pressing Return without any input doesn't change anything.

1.25 file deletion

Delete current file (Delete file)

If you decide that the current file is no longer needed you can delete it by pressing Amiga-Del. You will be asked whether it is really to be deleted. Press OK-gadget to confirm. The deleted file will stay in memory, and you can undo deleting by saving it again. But due to formatting there may be slight or big differences to original file. Don't try to recover a deleted executable in this way!.

1.26 %-display

%-Display (Switch %)

The display of text position in the status field can be switched between percentage and line number by pressing "%". If you are in %-mode the GOTO command will only accept %-values, otherwise line numbers.

1.27 archives

Archives (View archive, Flush tempdir, Unpack all/select)

For viewing and extracting archives the following programmes are needed:

archiver/unpacker	suffixes of archives
LhA 1.38e	.lha, .lzh, .run
LZX 1.21 changed!	.lzx
UnZip 5.12	.zip, .exe (see note)
UnRAR 2.04.1 (A.Kleinert)	.rar
GZip 1.2.4 (18.8.93)	.gz, .z, _Z, .tgz, .taz

You may use UnLZX (V1.21) and LHX instead of LZX and LhA respectively. Paths of archivers must be known to NEXT. I recommend to write their paths into ENVARC:Next.prefs. Otherwise NEXT will prompt for a path when needed. For TAR archives no external unpacker is required.

All (de)archivers mentioned above (except GZip and UnRAR) can be made resident. If you choose the option RES this will be done automatically by NEXT when an archiver is needed the first time. I recommend to do this if you have enough memory. The performance of archive extraction will increase remarkably.

NEXT recognizes archives by internal signs, not by the name suffix. After an archive has been detected a list of its contents will be created and displayed in a window. Moving through the list and selecting a file is described above.

The title of the list window gives some information about the archive. Most of it is selfexplaining, except the number in [] behind the name. It indicates the archive type: 1=lha 2=lzx 3=zip 4=gzip 7=tar. Normally the type is indicated by the name suffix, but this isn't always the case.

A selected file will be extracted to TEMPDIR/x and displayed afterwards. If the selected file happens to be an archive again another list will be created.

Note about ZIP archives

Zip archives for PCs are quite often self-extracting. Mostly they have the suffix .exe and thus cannot be recognized by name. NEXT now offers the possibility to check MSDos/Windows executables (except DLLs) for archives without need to load them. If you try to load an executable a requester will appear with 3 choices:

1. Check for zip archive (OK)
2. Load file (Load)
3. Cancel (Cancel)

Pressing OK will start UnZip and display the archive contents if it is an archive. Otherwise you'll get an error message.

There are 5 functions dealing with archives:

View old archive (V) displays the contents of last archive selected and enables extraction of files as described above.

View (from) readme

This feature is especially useful for AMINET archives. If you have loaded a .readme file press Shift-V and a list of files within the corresponding archive will be presented. If there is no archive but another file which corresponds to the .readme you'll be asked whether to load this one.

Flush tempdir (Shift-Del) deletes all temporary files in TEMPDIR/x if there are any. So you'll get back the memory or disk space they occupied. A requester will pop up if there are any files to delete and inform you about the number of files and their total length in Kb.

Unpack all (U) extracts all files within the current archive to a directory which must be specified by you. Paths of archived files can be preserved by clicking on gadget "whith path".

Unpack select (2xV) enables unpacking of one or more selected files. You get into select mode by pressing "V" two times. The first "V" opens the list window and displays the archive contents (if an archive has been read before). The second "V" switches into select mode. Filenames are now displayed with color 2. Select the files you want to extract by a click. A second click will unselect a file. After selection press Return and enter the destination directory.

You can unpack an archive without loading a file before. The main window of NEXT must not be open. Simply type >next <archive>. Press Return to extract the whole archive after the list window appeared. You'll be asked for the destination directory, then. Extraction of selected files works as described above.

1.28 tabs

Tabs (Tab width)

After pressing the TAB-key a new tab width can be entered. The change will be visible immediately if the current text contains tabs. It may be necessary to reload the file in order to adjust the line length according to the new width. After reloading NEXT displays the same text position as before. For prints tabs will be expanded by spaces if tab width is different from 8.

1.29 megafiles

Megafiles

If a file is too large to fit into one memory block (that's while I named it "megafile"), NEXT will split it into several parts and use half of the largest memory block available in this case. XPK- und powerpacked files can't be splitted.

Within a splitted file you can quickly move to next or previous part by repeatedly pressing the cursor keys (<- and ->).

NEXT will automatically switch to next/previous part if you are at top or bottom of current part and you use one of the MOVE commands (LINE up/down, PAGE up/down).

SEARCH, COUNT, GOTO, PRINT are allways restricted to the current part. This may change in a future version.

1.30 settings

Settings

The settings window is opened by "" and looks like this:

```
+-----+
| Settings          | Effect when switched on:
+-----+
| 1 ALIB           | display of guides by amigaguide.library
| 2 PURE           | no file conversion
|~~~~~F FLUSH~~~~~| Deletion of temp. file on exit
|~~~~~A ASK~~~~~  | request when scanning a directory
|~~~~~M MAKELINKS~~~~~|
| P PAUSEPRINT    | pause after each printed page
| S SMARTPRINT    | "smart" print mode
| K BOOKMODE      | page numbers placed like in books
| O ODDSFIRST     | odd pages are printed first
| C CENTERTITLE   | heading centered
| B BOLD          | heading bold
|~~~~~U UNDERLINED~~~~~| heading underlined
| I ITALIC        | heading italic
+-----+
```

Active modes are highligthed. They can be deactivated by a click and reactivated by another click. You may use the key shortcuts in front of each mode, too. The changes will be saved by pressing Return. You can quit without saving changes with Q, ESC and Space.

1.31 editing current file

Editing the current file (Edit file)

By pressing Shift-E an external editor can be started with the current file as argument. The path of the editor can be defined by option ED,

either on shell command line or in ENV:next.prefs. Otherwise you have to enter the path the first time you call this function. The editor is started asynchronously and gets a stack of 6000. I hope this will be sufficient in all cases.

If the current file is a pure text file the editor will be called with its path. If the current file is binary, compressed or already deleted you'll be asked if you want to edit the current (formatted) buffer. A positive response will NEXT cause to copy the buffer to TEMPDIR/<file> and then start the editor with this path as argument.

You can easily reload the edited file with "E" (no Shift this time).

1.32 copy files

Copy files

This command works like c:copy. But it only copies files which have been previously viewed by NEXT. After pressing Shift-Y a list window will open with these files. After selection of a file you'll be prompted for the destination. This may be either a filename or a directory.

In contrast to PRINT file this command copies the original file and not the more or less modified buffer of NEXT. The sense of Copy file is that you don't need to quit NEXT if you want to get a useful program or text file on your hard disk.

1.33 font selection

Font selection (Select font)

The font requester of reqtools.library can be called by pressing Shift-F. If the selected font is of different width than the old one, the current file will be loaded again, otherwise the new font will be used immediately.

1.34 reformat

Reformat current text (Reformat)

When you press Shift-R NEXT tries to reformat the current file "intelligently". ←
How this works is best demonstrated by this chapter which contains some very long lines. These lines will be wrapped according to window width and the following lines will ←
be appended,
replacing linefeeds by spaces. Now press Shift-R and wait...
This is the result.

Reformatting will not be done with binary files and it does not act on indented paragraphs.

1.35 changes

Changes

meaning: + new, ^ improved, * changed, - removed

V3.01

=====

This is only a minor update of Next3.0. It contains mostly bugfixes.

meaning: + new, ^ improved, * changed, - removed

- + Better detection of PBM and XBM pictures, some of them were displayed as TEXT.
- + NEXT displayed wrong file sizes for very big archived files (> 9.5 M) due to restriction to 7 digits. Now a "+" is appended to signalize that a file is bigger than reported.
- + Message "Extracting <file>" in the titlebar when extracting archives.
- * Help: The order of commands has slightly changed.
- * NEXT doesn't clear the screen title any longer.
- BUG: Next sometimes displayed an additional line (rubbish) when the displayed file didn't fill the window and the bottom command was performed.
- BUG in MatchFiles: The busypointer didn't disappear if no file had been found.
- BUG: MyFiles didn't display all files if ENV:next.files had no comment at the beginning. Caused by an "improvement" :-(
- BUG? Raphael Pilarczyk reported MuForce hits when the reqtools filerequester was opened or closed. I couldn't find the cause because there are no hits on my system. According to the output of MuForce the hits are produced by reqtools. I changed the code a little bit hoping that MuForce will stay quiet now.

V3.0

=====

- + Conversion of Unix/Linux "man pages" with nroff formatting instructions.
File type in Info: MAN
- + Conversion of WordPerfect Documents (V5-V7, for Windows).
File type: WPD
- + Conversion of IFF text.
File type: FTXT
- + Conversion of FinalWriter/FinalCopy and WordWorth documents.
File types: FW and WW.
- + New function "MatchFiles". Works like the AmigaDos command Search with options FILE ALL. See chapter Load for details.
- + Documentation for NEXT now in AmigaGuide format.
- ^ Next is now aware of the picture.datatype V43 and displays 24bit-pictures if subdatatypes allow it.
- ^ Information about a picture are now shown by a requester on the pictures screen (no screen switching). The requester also has a "Print" gadget.
- ^ Guides: Line numbers within LINKs are now supported.
- ^ Guides: More support for colors (@{FG/BG/APEN/BPEN...})
Color 3 is reserved for LINKs.
- ^ Headings of Unix man pages containing backspaces are now displayed correctly. The following lines show the difference:

Next29: S S S SY Y Y YN N N NO O O OP P P PS S S SI I I IS S S S
Next30: SYNOPSIS
^ Message "No Match" after search has been replaced by "No <string>"
^ Some smaller improvements concerning conversion of AmigaGuides, RTFs,
and HTMLs.
* New DTMode pointer.
* Renamed options: @LIB to AGLIB, @PUR to PURE. Please adapt next.prefs
if necessary.
* Settings: Renamed PAUSE to PAUSEPRINT and LINKS to MAKELINKS
- Option @CON has been removed because it's the default and opposite of
PURE.
- HQX-Decoding and listing of StuffIt archives have been removed.
I suppose nobody needed these features.
- BUG: Screen dump printed the whole screen not the picture.
- BUG: MyFiles didn't work with filenames set in "".
- BUG: WORD documents hadn't been detected in rare cases. (Hint from
Andreas Neumann)

1.36 thanks

Thanks to

- Nico Francois for publishing his decrunch routine and reqtools
- Charlie Gibbs for A68k. NEXT has been assembled with it.
- Matt Dillon for DME
- Hajo Stengert for betatesting and many valuable suggestions
- Andreas Neumann for uploading every new version of NEXT into Aminet
and providing me with information on file formats. Thanks also for
bug reports and useful suggestions!
- Georges Goncalves for new ideas, most of which are now implemented
- Matthias Blumenthal and Andreas Koch for their nice reviews

Thanks also for postcards and letters to

- Carl Barry
- Marco de Vitis
- Fred Küchenmeister
- Szutko Wlodzimierz
- Claude Lavigne
- Firmin la Huit
- Gustavo Gomez
- Marc Schilg
- Rand Fredricksen
- Stuart Caie

Bug reports, suggestions, postcards (could be more) may be sent to the
following address:

Jürgen Klawitter
Südenstr. 28
D-12169 Berlin
Germany