AmigaElm

Andreas M. Kirchwitz

	COLLABORATORS		
	TITLE : AmigaElm		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY	Andreas M. Kirchwitz	December 7, 2024	

	REVISION HISTORY		
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

AmigaElm

1.1   AmigaElm Help

Amiga Elm v8, Copyright © 1991-1995 by Andreas M. Kirchwitz

Presentation What is Amiga Elm Shareware Notice Elm is not gratis Introduction Too hot to handle Command Keys Sticky fingers Built-in Pager Quick and fast Send-off Window Last chance... before take-off Configuration File Build individual configuration Command Line Options Start Elm from CLI Additional Features More and more... Aliases Aliases MIME (Multimedia) Multipurpose Internet Mail Extensions Security (PGP) Encryption and digital signatures Internet, TCP/IP Elm with AmiTCP, AS225 etc. Installing How to install Elm External Programs Useful extra utilities Miscellaneous Bugs, ToDos, Infos, Thanks ... Disclaimer And now for something completely different Author Info About the Author

"With AmigaElm you're always ahead of your time."

-- Ralph Babel (author of "The Amiga Guru Book")

"AmigaELM is one of the very first mail readers that has in-built support for cryptographic applications, namely PGP. Henceforth, AmigaELM improves your privacy and general living quality significantly. An extraordinarily fine piece of software."

-- Peter Simons (author of AmigaPGP)

1.2   AmigaElm/Shareware Notice

I simply can't afford working on Elm just for fun...

This program is SHAREWARE.

As you know, this means that I would like to have some bucks if you use it frequently. Something like US $20+ (emphasis on the PLUS :-) will get you the latest updates of "Elm" directly from me (e-mail only) if you have no access to Usenet (alt.sys.amiga.uucp.patches) or Internet (Anonymous FTP), as soon as they are ready (more money probably means faster development). Anyway, you will be informed via e-mail when new versions are available. Of course, you also gain my favor - suggestions for improvements will be taken much more seriously from registered users.

The best (=cheapest) way to send money is cash. If this is illegal in your country (even for small amounts of money) I can offer you my bank account (giro transfer account). Notice: normally, remittance is very expensive across countries!

Name of bank : Deutsche Bank (Germany) Bank code : 100 700 00 Account number: 4701751

Checks and postal money orders must be payable in "DEM" (Deutsche Mark). Checks are ONLY accepted from within Germany. DON'T send checks from outside Germany. DO NOT!

For my e-mail/postal adddress refer to the Author Info. And please, include YOUR e-mail address if you send me letters.

The binary of the evaluation version maybe has some annoying things in it, so if you use this program on a regular basis, you really should register. The registered version of Elm comes with my support and full C source.

Actually this evaluation version is not crippled in any way. I trust YOU, that you pay for it if you use it. If this does not work -- then future versions of AmiElm will be crippled, definitely.

1.3   **AmigaElm/Presentation**

This piece of software is Shareware.

AA ii EEEEEE lll AAAA EE EE ll AAAA mm mm iii ggg gg aaaa EE ll mm mm AA AA mmm mmm ii gg gg aa EEEE ll mmm mmm AAAAAA mm m mm ii gg gg aaaa EE ll mm m mm AA AA mm mm ii gggg aa aa EE EE ll mm mm AA AA mm mm iiii gg gg aaa aa EEEEEEE llll mm mm ggggg

Amiga Elm, Copyright © 1991-1995 by Andreas M. Kirchwitz. == All rights reserved.

electronic mail: <elm-fan@zikzak.in-berlin.de> or <elm-fan@zikzak.snafu.de>

based on hwr-mail by Heiko W.Rupp (hwr@pilhuhn.ka.sub.org) ---

Consequently, after becoming more and more disenchanted with the existing electronic mail facilities (dmail, amr, avm, am) I decided to undertake creating my own system.

In any case, the system should be useable without reading any of the documentation -- but you'll miss lots of nice features!

Any comments or problems with any of the documentation or the program itself are welcome; if you can't get electronic mail to me, please feel free instead to drop me a note via the overland mail address given below.

Amiga Elm was written with the "CygnusEd Professional" editor and successfully compiled with the "SAS/C 6.55" (old Lattice) compiler.

1.4   **AmigaElm/Introduction**

The Index (Main) Screen

Upon entry to the program, the main screen will be displayed as below:

Mailbox is 'uumail:amk' with 15 messages [ELM3, SCREEN AmigaElm.1]

-> N 1 Apr 24 Larry Fenske (49) Hello there N 2 Apr 24 jad@hpcnoe (84) Chico? Why go there? D 3 Apr 23 Carl Smith (53) Dinner tonight? NU 4 Apr 18 Don Knuth (354) Your version of TeX... N 5 Apr 18 games (26) Bug in cribbage game AF 6 Apr 15 kevin (27) More software requests 7 Apr 13 John Jacobs (194) How can you hate RUSH? U 8 Apr 8 decvax!mouse (68) Re: your Usenet article L F 9 Apr 6 root (7) O 10 Apr 5 root (13) (none)

!=shell,l=pipe,+,-,<,>,?,help,<n>=set current to n c)hange folder,C)opy,d)elete,e)dit,f)orward,g)roup reply,ˆG)oto last msg, m)ail,o)ptions, e(x)it

Command: @

There are a number of things to notice about this, the main screen.

Most likely, on your computer the message currently "active" will be displayed in an inverse bar rather than being delimited by the two character arrow as here. It's nothing to worry about; displaying inverse video is just quite difficult on printable guides!

The first line on the screen (the screen title) always displays the name of the current folder, the number of messages in the folder, the current Elm version number, and the Elm public screen name.

The arrow ("->") or inverse video bar will always indicate which is the current message.

The first field that appears associated with each message is the status field. This can be blank (as with most of the ones above, or can have any combination of the following:

These characters signifies temporary status:

N for a new message O for an old (i.e. not new but not read) message D for a deleted message

These characters signifies permanent status:

U for urgent mail L for a logged message (i.e. written to a file or printed) F for a message that has been forwarded A for an answered (replied) message B for a bounced message

Continuing from left to right, the next field is the message number. For the most part you can ignore these unless you want to quickly move to a specific message (as we'll see later).

The date associated with each message is typically the date the person actually sent the message.

The next field displayed indicates whom the message is from. Elm will try to display the full name of the person who sent the message, rather than the their return address or computer login. Some systems don't generate the correct headers, though, hence messages like numbers 2 and 8, where it's their return address.

The number in parentheses is the total number of lines in the message (without header).

The final field is the subject of the message. Notice that messages might have an empty or not have any subject, as in messages #9 and #10.

The maximum of messages displayed depends on the screen dimensions and size of the default system font. Further into the document we'll learn how to change "pages" in the folder.

The three line Menu display will always indicate the relevant commands. Some commands do not show up on the mini-menu because they are somewhat esoteric, but they do appear on the '?' help screen (can you find it there, though?).

Finally, the '@' character indicates where the cursor would be, awaiting your input.

The typical action at this point is to use the <return> or <space> key to read the current message, which will clear the screen and display the current message:

Message 1/15 from Larry Fenske Apr 24 '87 at 8:45 pm

Hello there Dave, Just wanted to drop you a brief note to see what was going on with you this afternoon. Life here has been the usual fun and games... Ah well, off to the great wilds beyond the desk! Larry

1.5   AmigaElm/Command Keys

Before we go further with our example, however, let's look at all the functions available from the main screen:

!=shell,l=pipe,/=search,+, -, <, >,?,help,<n>=set current c)hange folder,C)opy,d)elete,e)dit,f)orward,g)roup reply,ˆG)oto last msg, m)ail,o)ptions,p)rint,q)uit,r)eply,s)ave,u)ndelete,w)rite,or e(x)it

That's not all, we have some menus for most of the functions. Menus are activated by pressing the right mouse button down... naah, read your AmigaOS manual for that trivial stuff ;-) Of course, OS 2.0 only, hehe.

Folder Message Mail Page Miscellaneous

Change Display Reply Next Shell Reload Goto... next Group reply Previous Pipe Resync previous Extend. g.r. Redisplay Palette Edit first Forward Alias Iconify number Bounce Options About pattern Send new List commands Quit Tag/untag Metamail Extra Program Exit Untag all Save Save with header Print Delete Undelete Mark unread Decrypt

Shifted keys

Use the shifted command keys for reply, forward, bounce and group reply to get the whole message (with header) included in your message. By default only the message body (text) is included. Use shift for exit/quit to exit/quit quickly (without any confirmations).

1.6   Command Keys/Information

Key: <help>

If you have installed the Amiga Guide System correctly, Elm's manual will be loaded (with section "Command Keys" displayed). Amiga Guide is a help and on-line documentation system for the Commodore-Amiga computer. This package is available from CATS.

AmigaGuide needs to know where to find your databases (help files, documents etc); for that the environment variable "env:AmigaGuide/ exists. This variable contains the list of directory names that AmigaGuide will search through when it attempts to open a database. The directory names are separated by a space. Be sure that the unmodified "Elm.guide" file is located in one of these directories.

1.7   Command Keys/Iconify

Key: i, Right-Amiga i

If Elm runs in its own window, the window will be iconified (if currently not iconified) or un-iconified (if currently iconified).

In fact, when iconified you can use any key or a click with the right mouse button to uniconify Elm's window. Really easy, eh? ;-)

Note: the "iconify" (zoom) gadget has gone because it never worked reliable. Sorry for that. Maybe it comes back in future releases.

1.8   Command Keys/Version

Key: v

Some information about Elm (version), the Author and a Shareware notice. (needs reqtools.library)

1.9   Command Keys/List of commands

Key: ?

A short overview of all the possible commands and a very short description of each command.

1.10   Command Keys/Movement

Keys: +, j, J, <cursor-down>, 2 (keypad)

Move the current message pointer down to the next message (going to the next page if needed). Use 'J' to skip over deleted messages.

Keys: -, k, K, <cursor-up>, 8 (keypad)

Move the current message pointer up to the previous message (going to the previous page if needed). Use 'K' to skip over deleted messages.

Keys: <, <cursor-left>, 4 or 9 (keypad)

Goto previous page.

Keys: >, <cursor-right>, 3 or 6 (keypad)

Goto next page.

Key: ^G (ctrl-g), ^E (ctrl-e), 1 (keypad)

Move the current message pointer to the last message in the folder. What the last message is depends on the sorting criterion.

Key: =, ^A (ctrl-a), 7 (keypad)

Move the current message pointer to the first message in the folder. What the first message is depends on the sorting criterion.

The cursor keys will only work with AmigaOS 2.0 (or higher). Movement via keypad works on all Amigas.

1.11   Command Keys/Redisplay

Key: <ctrl-l>

Redisplay the index screen. This should never be necessary, and it's a relict from my alpha tests for version 1.09 when running Elm on the workbench window and using the "Zoom" gadget. Redisplay is done automatically by default. But never say never :-)

1.12   Command Keys/Program

Key: \

This allows you to execute a predifined command without leaving the program. For historical reasons the default is a new shell ("newshell", see .elm/elmrc Shell) is fired up.

With AmigaOS 2.0 (or higher) you can make Elm's screen public to let newshell (or any other program) open its window on Elm's screen rather than on the Workbench screen. If so, you have to quit the shell before leaving Elm.

1.13   Command Keys/Search

Key: /

You're prompted for a DOS pattern (see your Amiga DOS manual). Starting with the current message Elm searches the mail folder (line per line) for that pattern. If any line matches your pattern Elm sets the current message to the first message where the pattern was found.

For example, the pattern "From: #?wusel#?" matches the first message sent to you by someone whose address contains the string "wusel".

You'll be informed if the pattern does not match at all.

For some special patterns Elm does an optimized search (no access to folder). These patterns have to start with header keywords like "From: <your pattern>" or "Cc: <another pattern>". This optimized search is very fast, but has a minor bug (some say it's a feature): those optimized patterns with header keywords only match occurences of that pattern in the header. Matches in the mail body are not found. And never forget the space between header keyword (after colon) and pattern!

1.14   Command Keys/Shell

Key: !

This opens a new shell on Elm's screen. There's no need to make the screen public. Handles also multiple shells (with adapted dimensions and position) at the same time.

Only works with AmigaOS 2.0 or higher.

1.15   Command Keys/Pipe

Key: |

This command allows you to pipe the current message through a given command which defaults to PipeDefault from your .elm/elmrc file.

On every invocation you are asked to accept the default or to modify the command. Elm remembers any modifications, and on the next call it offers you the previously entered command. Once you cancel this command Elm restores the pipe command to its default.

The pipe command may contain exactly one "%s" (or %S) which will be replaced by the name of a temporary file. For example:

You enter ... What Elm does ...

"uudecode %s" "uudecode t:a_temporary_file" "uudecode -abc <%s -xyz" "uudecode -abc <t:a_temporary_file -xyz" "uudecode" "uudecode" (Input() = t:a_temporary_file) "uudecode %s -xyz %s" "uudecode %s -xyz %s" (Input() = t:a_temporary_file)

It's not really a pipe. Elm copies the message into a temporary file and calls the user-defined command with that (absolute) filename as an argument or on standard input.

If you have tagged one or more messages Elm joins all the tagged messages together into one big (temporary) file and calls the pipe command on this file. Use "%S" (uppercase) instead of "%s" (lowercase) to call the pipe command on each message separately.

Only works with AmigaOS 2.0 or higher.

1.16   Command Keys/Display the current message

Keys: (using a user-defined pager) <return>, <space>

When issued from the index (main) screen, it will display the first screen (page) of the current message; and then when issued while in the built-in pager (prompting "--More--" or something like that), it will page through the message to the end; and then when issued at the end of a message (when "End of Message" appears), it will bring you back to the index screen.

If you're using an external pager Elm starts it synchronously. While the pager is running, you cannot make any input to Elm. Quit your pager after reading the message and Elm jumps back to the index screen automatically. If your pager detaches itself from the cli -- BE CAREFUL.

WARNING:

Do not use any of Elm's functions before the pager has terminated.

1.17   Command Keys/Goto first message of folder

Key: =

Set current message to first message of folder.

Note that changing the current message to a message not on the current page of headers will result in a new page being displayed.

1.18   Command Keys/Specify new current message

Key: <digit key/s>

When you type in any digit key Elm will prompt "Set current to : n", where 'n' is the digit entered. Enter the full number and terminate with <return>.

Note that changing the current message to a message not on the current page of headers will result in a new page being displayed.

1.19   Command Keys/Re-read folder

Key: @

Under rare circumstances it might be useful to re-read the current folder. Note: the current contents of the folder and the current status flags are not saved. Of course, the current status flags do not go lost on re-read.

Elm re-reads the folder automatically when it has been modified. Under AmigaOS 2.0 (or higher) Elm uses file notification to detect modification. This way the folder is re-read also if no user interaction is taken. Elm has a built-in delay of about 60 seconds before re-reading the folder to avoid trouble with a currently running sendmail which is delivering lots of mails.

If the user interacts on the index screen, Elm re-reads the folder immediately if it has changed. (AmigaOS 1.3 and 2.0)

1.20   Command Keys/Resynchronize folder

Key: $

On more advanced systems, it's nice to start up the mailer in a window and let it sit in background unless new mail arrives which point it can be brought up to the forefront of the system and read.

In this case, the modified mail folder is saved not very often, so when your system chrashes all modifications since last (re-) read of the folder are lost. To avoid to quit and start Elm again, this command lets Elm write the current folder back to disk (like "Quick quit", see below) and read it again (like it was started again).

1.21 Command Keys/Edit aliases

Key: a

This allows you to modify the current alias file (with a user configurable editor) at a single keystroke. Elm re-reads the alias file if it has been modified.

1.22 Command Keys/Bounce mail

Key: b

This "remails" mail to someone else in such a way as to make the return address the original sender rather than you. (The forward command is similar, but it makes the return address you rather than the original sender.)

Elm asks you for one or more destination addresses. Addresses are separated by a comma. Neither you can edit bounced mail nor you can append a signature to the mail.

1.23 Command Keys/Copy to folder

Key: C

This command copies the current message or the set of tagged messages "as is" to a folder. If there is anything in the folder currently the message or messages are appended to the end, otherwise the folder is created containing only the newly copied message(s). The prompt for this command is `Copy to folder: '. A response of <return> cancels the command and returns the user to the system prompt. The usual filename metacharacters are available, too. That is, this command expands filenames with '=' (on first position) being your maildir directory ("uumail:" by default); i.e. "=dave" is expanded to "uumail:dave".

If you have installed the asl.library or reqtools.library, you can enter the name of the destination folder with a comfortable string requester. The default (starting) directory is "uumail:".

See also commands Save and Write.

1.24 Command Keys/Change folder

Key: c

Specifying this command allows the user to change the folder that is currently being read. This is intended for perusal and reply to previously archived messages. The prompt is "Name of new folder: " and entering <return> cancels the operation, while entering a filename causes the program to read that file as the new folder, if possible. The usual filename metacharacters are available, too. That is, this command expands filenames with '=' (on first position) being your maildir directory ("uumail:" by default); i.e. "=dave" is expanded to "uumail:dave".

If you have installed asl.library or reqtools.library, you can select the name of the new folder with a comfortable file requester. The default (starting) directory is "uumail:" and can be changed with the config variable FolderDirectory in .elm/elmrc.

1.25 Command Keys/Delete and Undelete

Key: d and u

Neither of these two commands have any prompts and indicate their action by either adding a 'D' to the current message index entry (indicating deletion pending) or removing the 'D' (indicating that the message isn't set for deletion).

1.26 Command Keys/Edit mailbox

Key: e

This allows you to modify the current mail file (with a user configurable editor) at a single keystroke. This is mostly useful for editing down messages before saving them. Modifying headers should be done with extreme caution, as they contain routing information and other vital stuff for full functionality.

Note that Elm re-reads the folder if the folder has been modified; this operation can take a little while.

1.27 Command Keys/Forward

Key: f

Allows the user to forward the current message to another user. This copies the message into the edit buffer and allows the user to add their own message too. (See also bounce, above.)

1.28 Command Keys/Group reply

Key: g

Identical to reply, except that the response additionally is mailed to all other recipients (To, Cc) of the original message (except yourself).

Use Extended group reply if you want the recipients of the "From:" line to be included. This could be useful when replying to mails from mailing-lists.

1.29 Command Keys/Extended group reply

Key: E

Identical to reply, except that the response additionally is mailed to all other recipients (To, Cc, From) of the original message (except yourself).

This could be useful when replying to mails from mailing-lists, because then "Reply-To:" is set to the mailing-list and not to the author of the mail. Don't use this function for group-reply on "normal" mail.

Use Group reply if you don't want the recipients of the "From:" line to be included.

1.30 Command Keys/Display header

Key: h

This command displays the current message with all headers intact. When you display a message with other commands, certain header lines are formatted and others discarded.

1.31 Command Keys/Mail

Key: m

Send mail to a specified user. The editor is invoked and the message can be composed.

1.32 Command Keys/Metamail -- MIME

Key: M (followed by 'r' or 'g')

Small hack for a "multimedia mail reply" to the sender of the current message. You can choose between normal (r, R) and group reply (g, G).

1.33 Command Keys/Unread

Key: n

Mark message as unread. If the message was new for this session the 'N' flag will appear again; else the message is marked as old (not read yet but not new for this session).

1.34 Command Keys/Options

Key: o

This full-screen display allows you to alter the settings of a number of parameters. It's self-documenting (where have you heard that before?) so isn't explained in too much detail here.

1.35 Command Keys/Print

Key: p

This allows you to print out (prt:) the current message.

1.36 Command Keys/Color Palette

Key: P

Allows you to change screen colors (if reqtools.library is installed). Changes are temporary and thus they cannot be saved.

1.37 Command Keys/Quit Elm

Quit, key: q ----

Quit Elm altogether. If any messages are marked for deletion, it will ask if you want them deleted. If there are read messages (not marked as new or old) in your folder, Elm will ask if you want them kept in the incoming mailbox.

You can abort the questions - and thus, abort the quit function completely - with pressing Control-C or the Escape-key.

Quick quit, keys: Q, ESC ----------

This behaves similar to the 'q' command except that you are never prompted for answers to the message disposition questions. By default, if any messages are marked for deletion, they will be deleted without further inquiry. Read messages are kept in the incoming mailbox.

The default behavoir depends on AlwaysDelete and AlwaysKeep from your .elm/elmrc file.

1.38   Command Keys/Reply

Key: r

Reply to the sender of the current message. A copy of the source message is copied into the edit buffer prepended with the prefix character sequence specified in your .elm/elmrc file.

1.39   Command Keys/Save to folder

Key: s

This command is like the Copy command (see above), except that the saved message is decoded (MIME, if necessary) so that it is human readable.

In opposition to the Write command, the header of the message _is_ written to the specified file. Because of the automatic decoding, the (unmodified) header may not reflect the actual contents of the message body correctly; so files created with this command cannot be used by Elm afterwards.

This command expands filenames with '=' (on first position) being your maildir directory ("uumail:" by default); i.e. "=dave" is expanded to "uumail:dave".

1.40   Command Keys/Tag message

Key: t

Tag (or untag) the current message for a later operation. Currently only print, write and save support this. Elm processes tagged messages in the order they were tagged.

See also Un-Tag messages.

1.41   Command Keys/Un-Tag messages

Key: T

Untag all previously tagged messages.

1.42   Command Keys/Write to file

Key: w

This command is like the Copy command (see above), except that the saved message is decoded (MIME, if necessary) so that it is human readable.

In opposition to the Save command, the header of the message is not written to the specified file; so files created with this command cannot be used by Elm afterwards due to the missing header.

This command expands filenames with '=' (on first position) being your maildir directory ("uumail:" by default); i.e. "=dave" is expanded to "uumail:dave".

1.43   Command Keys/Exit Elm

Exit, key: x ----

This leaves Elm discarding any changes to the mailbox. If changes are pending (such as messages marked for deletion) a prompt is made to confirm discarding the changes. If confirmed, no messages are deleted and the statuses of messages are unchanged.

That is, any messages that were new will remain new instead of being noted as old, and any messages that were read for the first time will be again noted as unread.

You can abort the question - and thus, abort the exit function completely - with pressing Control-C or the Escape-key.

Quick exit, key: X (or closing gadget) ----------

This leaves Elm in the quickest possible manner without even prompting about discarding the changes to the mailbox. No messages are deleted and the statuses of messages are unchanged. That is, any messages that were new will remain new instead of being noted as old, and any messages that were read for the first time will be again noted as unread.

1.44   Command Keys/Decrypt

Key: y

Decrypt current (or tagged) message(s) with predefined command for decryption.

See also Decrypt in .elm/elmrc.

1.45   AmigaElm/Built-in Pager

The Built-in Pager

Now, back to our example. The built-in pager shows you each page of the message; it deals also with lines longer than the actual screen width. The pager offers you some additional choices which are equivalent to the correspondending functions from the index screen: Mail, Reply, Forward, Delete. But there's one more feature: Header. This command displays the current message with all headers intact. When you display a message with other commands, certain header lines are formatted and others discarded. Choose Header again to display the message from the beginning in the old manner. (there's a flag in the configuration file for displaying headers; properly speaking, the Header command switches this flag)

If you want to redisplay the message from the beginning (like quit and then diplay) use the '^' key.

You can quit the built-in pager at any time with 'q' (quit) or 'i' (index). Both keys are equivalent.

With 'n' you skip directly to the next undeleted message. The main index screen will not be displayed, instead the built-in pager immediately jumps to the next undeleted message (if any exists, of course). With 'd' you delete the current message, return to the main index screen and set the cursor position on the next undeleted message. With 'D' (shift d) you delete the current message as well, but then you jump directly to the next undeleted message within the pager.

If you set SkipIndex in your .elm/elmrc file to "Yes", now commands like 'd' (delete) and ' ' (space) will skip the main index screen. And 'D' won't directly go to the next message within the pager but it will return to the index screen.

As you can see, SkipIndex inverts the meaning of 'd' and 'D' from within the built-in pager.

As said before, by default, only 'n' goes directly to the next undeleted message. Other commands like 'd' and ' ' return to the index screen and put the mark (horizontal inverse bar) on the next undeleted message.

Use the keys on the numeric keypad to browse through a file:

(or the index screen, respectively)

---------------------- |7 |8 |9 | | 1st| Line| Page| | Page| Up| Up| --------------------- |4 |5 |6 | | | | | | | <nop>| <nop>| <nop>| ----------------------- |1 |2 |3 | | Last| Line| Page| | Page| Down| Down| ----------------------

Equivalents...

First Page: ^ <Alt> <Cursor Up>

Last Page: e <Alt> <Cursor Down>

Page Up: <Backspace> <Cursor Left> <Shift Cursor Up>

Page Down: <Space> <Cursor Right> <Shift Cursor Down>

Line Up: - <Cursor Up>

Line Down: + <Return> <Cursor Down>

The scrollback buffer for page/line up is unlimited.

When you are on the last page of the message, <space> and <return> are also valid keys to leave the pager and to return to the index screen. While <return> stays at the current message, <space> is equivalent to 'n' and moves the mark (horizontal inverse bar) to the next message.

When you are about to send of a message under the forward, mail, or reply commands (see above), a small menu of the following options appears:

Forget

This gets you out of sending a message you started. If you are in send-only mode, the message is saved to the file specified in the configuration variable CancelledFolder in .elm/elmrc.

Edit

Entering this command will allow you to edit the text of your message again. At this point the .elm/elmheaders file is added to the text and all aliases and receipients of a group reply are expanded.

Send

This sends the message as is without any further ado, except that the signature (uulib:<user>.signature or uulib:.signature) is appended to the message.

Send without signature

This sends the message as is without any further ado.

Multimedia

Toggles between MIME-compatible header (RFC1521) and normal mail header (RFC822). Depends on MimeCoding for new mails. On replies Elm "clones" the coding mode of the original message. Also cycles thru all available Content-Transfer-Encoding modes.

BTW:

The file $HOME/.elm/elmheaders, uulib:<user>.elm/elmheaders or uulib:.elm/elmheaders is always appended to your message header.

1.46   AmigaElm/Send-off Window

When you're sending mail (mail, reply, forward, bounce) and leaving the editor then a nice window pops up under your mouse-pointer: the so called "Send-off Window". This breath-taking state-of-the-art GadTools-interface offers ultimate flexibility to the user and thus, it can be controlled with the keyboard as well as the mouse.

If you're running AmigaElm in terminal mode you'll see an ugly textual menu. Under Amiga OS 1.x you'll see a small ReqTools-requester instead of the GadTools-interface.

Neither the textual nor the ReqTools-interface will be discussed here in detail. Both are reduced versions of the GadTools-interface. So I only explain the absolutely KEWL Send-off Window with its GadTools-interface.

MIME Header, Content-Transfer-Encoding

Toggles between MIME-compatible header (RFC1521) and normal mail header (RFC822). Depends on MimeCoding for new mails. On replies Elm "clones" the coding mode of the original message. You can cycle thru all available Content-Transfer-Encoding modes.

Elm tries to choose the optimal Content-Transfer-Encoding mode.

(currently, 7-bit-encoding does not work)

Return-Receipt-To, <address>

If selected (defaults to RequestReturnReceipt in .elm/elmrc), then request sending of a return-receipt-to to the given address (defaults to ReturnReceipt in .elm/elmrc) when the mail arrives at the destination system.

Note: Some systems don't support Return-Receipt-To.

Return-View-To, <address>

If selected (defaults to RequestReturnView in .elm/elmrc), then request sending of a return-view-to to the given address (defaults to ReturnView in .elm/elmrc) when the mail is read by the user at the destination system.

Note: Most systems don't support Return-View-To.

Sent-Archive, <folder>

If selected, then save outgoing mail to the given folder (defaults to SentArchive in .elm/elmrc).

High Priority

If selected, then mark this message as urgent. It depends on the receipient's mail-reader if the appropriate action is taken to display this message as urgent.

Append Signature

If selected, then append the signature (uulib:<user>.signature or uulib:.signature) to the message. The default setting of this checkbox can be toggle with the variable AppendSignature in your .elm/elmrc file.

Security (none, Encrypt, Sign, Encrypt+Sign, Custom 1-3)

See Security and Encrypt / Decrypt in .elm/elmrc for details.

The following gadgets will exit the Send-off Window:

Okay

This sends the message as is without any further ado.

Edit

Entering this command will allow you to edit the text of your message again. At this point the .elm/elmheaders file is added to the text and all aliases and receipients of a group reply are expanded. Aliases are expanded for every edit-run.

Cancel

This gets you out of sending a message you started. If you are in send-only mode, the message is saved to the file specified in the configuration variable CancelledFolder in .elm/elmrc.

1.47   AmigaElm/Configuration File

The Configuration File

Elm, like lots of other software on the Amiga system, has the ability to automatically read in a configuration file at each invocation. The file must be called elmrc and reside in the .elm directory located in your uulib: directory. It can have any of the entries below, in any order. If you are missing any entries, or if you don't have an .elm/elmrc file, the default values (listed below for each option) will be used. Note that those options below designated with * can be altered using Elm via the o)ptions screen.

Elm tries to support multi-user; first it opens a configuration file named "$HOME/.elm/elmrc" ($HOME is a environment variable), if this fails it opens "uulib:<user>.elm/elmrc", where <user> is the name found in uulib:Config. If this all fails, Elm opens the file "uulib:.elm/elmrc" (<user> = "").

Same procedure is done with "uulib:<user>.signature", "uulib:<user>.elm/elmheaders", "uulib:<user>.elm/aliases" and all other user-dependent files.

The configurations file consists of a bunch of keywords and their values. Keywords begin in the first column and are seperated from their value by one or more white spaces (space or tabulator). It's not allowed to omit a value. Keywords and values are case-independent.

If you want to have white spaces in the value, surround it with double quotation marks. Thus, an empty string are two quotation marks with nothing between them. Please note, that it is NOT THE SAME to assign an empty string to a keyword and to omit a keyword completely.

There are lots of switches. "Yes" always turns a switch on (true), any other value turns it off -- of not explained otherwise.

For a better idea of how this all works, here's an example .elm/elmrc file with all keywords (and their defaults). While looking through it, notice that you can have lots of comments and blank lines for readability.

It's not recommended to use all keywords. Use only the ones which default value differs from your favorite setting.

Furthermore, do not rely on all default values. Most of them are out of date. They are made for beginners, and they make Amiga-Elm look like Unix-Elm.

Okay, now let's see what we have...

example .elm/elmrc - options file for the ELM mail system # (rename it to "<username>.elm/elmrc" for multiple users) # # Keyword ... My favourite value ...

general config # FullName Andreas M. Kirchwitz Editor StartCED AuxEditor Vim PrefixString "> " ReplyIntroduction Hi $FIRSTNAME ($NAME), in $MSGID on $DATE you wrote: ForwardIntroduction On $DATE, $NAME wrote: Signature uulib:.signature Pager builtin SigDashes No MimeCoding Quoted-Printable MetaMail Ask

SendMail / RMail # SendMail sendmail <$MSG RMail rmail <$MSG $TO FromHeader Yes NoMessageId No

AnswerReturnReceipt No AnswerReturnView No RequestReturnReceipt No RequestReturnView No ReturnReceipt malcolm@math.cha ReturnView hammond@jurassic.park

Internal # Buffers 20000 OldLocking No WBR_Timeout 500000 ReadContentLength No SendContentLength No WarnReturnCode 1

Look & Feel # ShowFullHeader No ShowSignature Yes Sorting Natural ReverseSorting Yes AppendSignature NoReply TopDownScrolling Yes SkipIndex Yes

SaveDirectory uucp:archive/text FilenameOffer USER FilenameBuffer No

Shell NewShell CON:////ElmShell/CLOSE/SCREENAmigaElm PipeDefault uucp:c/xuudecode PrintCmd copy % prt:

Folders # SentArchive uumail:mail.sent ReceivedFolder uumail:mail.received FolderDirectory uumail: AlwaysDelete Yes AlwaysKeep Yes KeepBackup Yes QuitEmpty No KeepEmpty Yes

Fonts # StdFontName topaz-classic.font StdFontSize 8 IntuiFontName helvetica.font IntuiFontSize 13 PagerFontName courier.font PagerFontSize 13

Intuition # UseOwnScreen Yes PubScreenName AmigaElm DisplayID "NTSC:High Res Laced" Width 1024 Height 1008 TopEdge 0 LeftEdge 0 ShanghaiWindows No PopPubScreen No DefaultPubScreen No WorkbenchToFront No ElmToFront Yes IconifyOnClose Yes IOWindow CON:20/20/640/200/AmigaElm-IOWindow/CLOSE/WAIT/AUTO

Colors # ColorBackground 15 12 11 10 ColorText 15 0 0 0 ColorIntuition 15 15 15 15 ColorPrompt 15 11 6 4

Security # Encrypt EncryptMsg $MSG $MODE Decrypt DecryptMsg $MSG CryptMode none

end of .elm/elmrc

1.48 Configuration File/IconifyOnClose

Default: No

If Elm is running in a window (and not on its own screen), it will have a close gadget which - by default - aborts Elm. If you set "IconifyOnClose" to "Yes", Elm won't abort but iconify when the close gadget is pushed down.

This is for all the people who miss the lost "zoom" gadget ;-)

1.49 Configuration File/SkipIndex

Default: No

When in built-in pager, by default, the key 'n' skips directly to the next undeleted message. The main index screen will not be displayed, instead the built-in pager immediately jumps to the next undeleted message (if any exists, of course). With 'd' you

delete the current message but return to the main index screen where the cursor position is set to the next undeleted message. With 'D' (shift d) you delete the current message as well, but then you jump directly to the next undeleted message within the pager.

If you set "SkipIndex" to "Yes", now commands like 'd' (delete) and ' ' (space) will skip the main index screen. And 'D' won't directly go to the next message within the pager but it will return to the index screen.

As you can see, "SkipIndex" inverts the meaning of 'd' and 'D' from within the built-in pager.

1.50   Configuration File/WarnReturnCode

Default: 10

When launching external programs (eg, sendmail), AmigaElm warns you (with a requester) if the return code (RC) of the program is "WarnReturnCode" or higher. If the RC is less "WarnReturnCode", AmigaElm ignores it and continues processing without warning.

1.51   Configuration File/CryptMode

Default: none

Set default encryption mode for outgoing mail (Send-off Window).

NONE : no security ENCRYPT : encryption SIGN : digital signature ENCRYPT+SIGN: encryption and digital signature CUSTOM1 : custom crypt mode #1 CUSTOM2 : custom crypt mode #2 CUSTOM3 : custom crypt mode #3

See also chapter Security.

1.52   Configuration File/FolderDirectory

Default: uumail:

Starting directory when changing folder (command key 'c') with file-requester.

1.53   Configuration File/WBR_Timeout

Default: 1

Set the timeout period (in microsends) for the connected terminal to respond to a WINDOW STATUS REQUEST (special Amiga control sequence to get window bounds). Only used in terminal mode, of course.

For local usage a small value is sufficient, for remote usage (eg, over a serial line) the value should be much higher (eg, 500000, which a 1/2 second).

1.54   Configuration File/Encrypt

Default: none

This is the filter program to encrypt an outgoing message (according to the security mode ($MODE) selected in the Send-off-window). The filter is responsible for encryption and replacing the original message ($MSG) with the encrypted one.

List of supported placeholders for Encrypt:

$MSG : name of file containing message (header + body) $MODE : "ENCRYPT", "SIGN" or "ENCRYPT+SIGN" and "CUSTOM1", "CUSTOM2", "CUSTOM3"

An example "c/EncryptMsg" (for use with PGP) is included in the AmigaElm distribution.

See also Decrypt.

1.55   Configuration File/Decrypt

Default: none

This is the filter program to decrypt an encrypted message (see command key for decryption, Decrypt key). The filter is responsible for decryption and display of this message on Elm's screen.

List of supported placeholders for Decrypt:

$MSG : name of file containing message (header + body)

An example "c/DecryptMsg" (for use with PGP) is included in the AmigaElm distribution. You should use a text-viewer (eg, MultiView) with the ability to save the text to a separate file (eg, in RAM-Disk) so that you can include it again later when replying to the mail.

See also Encrypt.

1.56   Configuration File/AnswerReturnReceipt

Default: No

If the sender has included a "Return-Receipt-To: <address>" in the message header, Elm will send back an acknowledge when the message was successfully read. Use with care, because normally "s(end)mail" handles it for you.

"Yes" turns this feature on, any other off.

1.57   Configuration File/AnswerReturnView

Default: No

If the sender has included a "Return-View-To: <address>" in the message header, Elm will send back an acknowledge if the message was successfully read.

"Yes" turns this feature on, any other off.

1.58   Configuration File/RequestReturnReceipt

Default: No

If you want to have an acknowledge when the receipient of a message has received this message, Elm adds a "Return-Receipt-To: <address>" to the message header, where address is the same like in "From: <address>".

"Yes" turns this feature on, any other off.

Use ReturnReceipt to specify a custom address.

Note 1:

Only works if the receipient's mail system supports it!

Note 2:

It's a very BAD habit to have this feature always activated. You should use it only if you really NEED the acknowledgement. People tend to ignore messages from other persons who are over-doing it...

1.59   Configuration File/ReturnReceipt

Default: $USER@$HOST$DOMAIN (same as in "From:")

Specify a different address for "Return-Receipt-To:".

See also RequestReturnReceipt.

1.60   Configuration File/RequestReturnView

Default: No

If you want to have an acknowledge if the receipient of a message has read this message, Elm adds a "Return-View-To: <address>" to the message header, where address is the same like "From: <address>".

"Yes" turns this feature on, any other off.

Use ReturnView to specify a custom address.

Note: Most mail systems do not support this feature!

1.61   Configuration File/ReturnView

Default: $USER@$HOST$DOMAIN (same as in "From:")

Specify a different address for "Return-View-To:".

See also RequestReturnView.

1.62   Configuration File/ShowFullHeader

Default: No

If you always want to see the full header of each message and not only some keywords (From:,Cc:,Bcc:,Subject:), set it to "Yes", any other value shows only a small reformatted header.

1.63   Configuration File/ShowSignature

Default: Yes

The message body (text) and the signature are divided by "-- \n" normally. Elm recognizes this and gives you the abiltity to hide signatures. If you want to see the signature, set value to "Yes", any other will suppress signatures.

1.64   Configuration File/PrefixString

Default: "> "

If you reply to a message, each line of the source text will be included in your new mail and prefixed with this string. If you do not want any prefix, set value to "". DO NOT OMIT THE VALUE!

1.65   Configuration File/SentArchive

Default: <none, do not save outgoing mail>

All outgoing mail is saved to a special archive. This archive is not meant to be a mail folder, so it should not be read by Elm. If you do not want to save outgoing mail, omit the value for this keyword or omit both (prefix line with '#').

1.66 Configuration File/ReceivedFolder

Default: "uumail:mail.received"

Where to save received and read messages to. If you quit Elm, all messages marked as read (not old, not new and not marked for deletion) may be saved to this folder. On exit Elm will prompt the user whether to save or not.

Elm does not move read messages to received folder if a folder name was specified via command line option. Instead messages will be kept in current folder.

See also AlwaysKeep.

1.67 Configuration File/AlwaysDelete

Default: Yes

If you quit Elm quickly ('Q'), all messages marked for deletion may be removed from the folder ("Yes") or not (any other value).

See also AlwaysKeep.

1.68 Configuration File/AlwaysKeep

Default: Yes

If you quit Elm quickly ('Q'), all read messages may be moved to the ReceivedFolder or may be kept in the incoming mailbox ("Yes").

See also AlwaysDelete.

1.69 Configuration File/FullName

Default: <ENV:Realname, and RealName from s:UUconfig/uulib:Config>

The full user name (real name) for outbound mail.

1.70 Configuration File/Editor

Default: "ed"

What editor to use for editing mails, folders etc. (if not in your DOS search path, use full (absolute) path). The editor should act synchronously (running in the foreground), that means it should not push itself to the background.

If you're using CygnusEd's activator "Ed", you must use the option "-s" or "-sticky". For TurboText's activator "Ttx", you must use the option "WAIT" to prevent the editor from detaching from the shell.

The string may contain exactly one "%s" which will be replaced by the name of the file. Warning: More than one "%s" crashes your machine!

This entry replaces the value read from "ENV:Editor" (if any).

1.71 Configuration File/AuxEditor

Default: none

This entry is similar to Editor.

If you start Elm in terminal mode, you might want to use a special editor which is useable via stdin/stdout, so that it works remote over a serial line. If "AuxEditor" is unset "Editor" will be used.

This entry replaces the value read from "ENV:AuxEditor" (if any).

1.72   Configuration File/SendMail, RMail, FromHeader

Defaults: SendMail sendmail <$MSG RMail rmail <$MSG $TO FromHeader Yes

There are several versions of the 'sendmail' program which is called by Elm. Each of them has a different behavior in parsing the mailfile and the command line options. For example: Ingo Feulner's sendmail has no -R (realname) option but it allows to omit the -f option and to put the complete "From:" line into the mailfile. Matt Dillon's sendmail has serious problems with "From:" within the mailfile, but it allows -R option.

SendMail is used when normal mails are sent. A "real" sendmail expects all informations within the header of the message (given on standard input).

RMail (or sendmail -r) forces sendmail not to process headers; the destination address is given as argument. The message (given on standard input) must not be modified. Elm uses RMail when "bouncing" messages.

FromHeader specifies if you wanna have a "From: ..." line in the message header. Normally, you want that. But some sendmails (like Dillon's) cannot handle that case, instead they need a from-address as argument (option -f).

Recommended settings for Matt Dillon's sendmail (AmigaUUCP D), wUUCP's sendmail and Michael B. Smith's sendmail (AU):

SendMail sendmail <$MSG -f $FROMUSER -R $FROMREAL RMail rmail <$MSG $TO FromHeader No

If you get problems with double realnames (you'll understand when you see it :) change the value for "SendMail":

SendMail sendmail <$MSG -f $FROMUSER

Recommended settings for Ingo Feulner's sendmail (AmigaUUCP Plus):

SendMail sendmail <$MSG RMail sendmail -r <$MSG $TO FromHeader Yes

Recommended settings for Michael B. Smith' INetUtils:

(see also AmigaElm & Internet)

SendMail SMTPpost $MSG RMail SMTPpost -r <$MSG $TO FromHeader Yes

Recommended settings for Kai Siering's sendmail (don't confuse this with wUUCP ;-), Ralph-Thomas Aussem's AmigaSMail and any other sendmail:

SendMail sendmail <$MSG RMail rmail <$MSG $TO FromHeader Yes

List of supported placeholders for SendMail:

$MSG : name of file containing message (header + body) $FROMUSER : user (account) name $FROMREAL : user's full realname (must not be quoted!) $FROMHOST : host name (without domain) $FROMDOMAIN : domain name (don't forget the leading dot!)

List of supported placeholders for RMail:

$MSG : name of file containing message (header + body) $TO : destination address

1.73   Configuration File/ReplyIntroduction

Default: "Hi $FIRSTNAME,"

If doing a reply, Elm puts an introducing line for you in front of the quoted text. To give this automated process a more personal touch, some useful keywords are available (relating to the original message):

$FROM : complete "From:" line $ADDRESS : e-mail address $USER : user (account) name $NAME : full real name $FIRST-NAME : only the first name $MSGID : message id $DATE : date $NICK : nickname (string between `` or ", like "wusel" in "Kai 'wusel' Siering")

If Elm cannot find the corresponding fact (for example, the real name is missing in the original message) the keyword remains unchanged. Oh no, that's wrong... sometimes there's a (internal) fallback to other keywords.

It's allowed to use more than one keyword, e.g.

"Hiya $FIRSTNAME, in $MSGID on $DATE you wrote:"

1.74   Configuration File/ForwardIntroduction

Default: "On $DATE, $NAME wrote:"

Same as ReplyIntroduction, but for forwarded messages.

1.75   Configuration File/KeepBackup

Default: Yes

Elm makes a backup of the current folder in uumail: (appending ".o"), before saving the folder to the original file. If you wish to keep this backup - even if no error occured while saving - set the value to "Yes", set it to any other value to delete the backup if the folder was written successfully to disk.

1.76   Configuration File/Color

Default: <workbench colors>

By default Elm uses the workbench colors (preferences) for its screen. You have to specify four colors (two bitplanes), which are equivalent to the ones given by the "palette" tool from Commodore's Workbench/Extras disks. Hexadecimal numbers must begin with "0x" (or "0X"), octal numbers with "0" and any other numbers are assumed to be decimal.

Example (my favourite color settings): 3258 0 0xFFF 2916

<keyword> <max> <red> <green> <blue> ColorBackground 15 12 11 10 ColorText 15 0 0 0 ColorIntuition 15 15 15 15 Color-Prompt 15 11 6 4

Convert old "Color" to new format:

For example, your old "Color" looked like this: 3258 0 4095 2916

This first number (3258) refers to "ColorBackground", the second (0) to "ColorText", the third (4095) to "ColorIntuition" and the fourth (2916) to "ColorPrompt".

The "max number" is always "15". Now for the RGB numbers. Divide (integer division!) the number by 256: 3258 / 256 = 12. This is your "red number". Subtract 12 * 256 from the old number 3258: 3258 - 12 * 256 = 186. Divide this by 16: 186 / 16 = 11. This is your "green number". Subtract "11 * 16" from 186: 186 - 11 * 16 = 10. This is your "blue number".

Same procedure for the other colors.

1.77   Configuration File/Buffers

Default: 10240

By default, Elm's buffer size for file I/O is 10240 bytes (10kB). To speed up saving a little bit, you may allow Elm to allocate an extra buffer (which has nothing to do with standard file I/O). This extra buffer is used to copy the message body from the old to the new folder. Therefore, this buffer should be large enough to hold most of your message bodies completely, so that the body can be copied in one part. This speeds up saving significantly.

Minimum buffer size is 1024 bytes.

Because of C's limitations ("int" for "fread()/fwrite()") buffers should not be greater than ca. 32000 bytes.

1.78   Configuration File/StdFontName

Default: topaz.font

Elm uses the "standard font" for the index (main) screen. To handle large subjects and as much messages as possible on one page you can choose a small font. At least 80x20 characters must fit on the screen.

Do not forget the trailing ".font" !

See also StdFontSize.

1.79   Configuration File/StdFontSize

Default: 8

Size of the "standard font". Under AmigaOS 2.0 (or higher) there need not exist a suitable file in the fonts: directory. The font will be scaled if needed.

See also StdFontName.

1.80   Configuration File/IntuiFontName

Default: topaz.font

The "intuition font" is used for window titles, requesters etc. (see standard font for details)

See also IntFontSize.

1.81   Configuration File/IntuiFontSize

Default: 8

Size of the "intuition font".

(see standard font for details)

See also IntFontName.

1.82   Configuration File/PagerFontName

Default: topaz.font

The "pager font" appears within the built-in pager.

(see standard font for details)

See also PagFontSize.

1.83   Configuration File/PagerFontSize

Default: 8

Size of the "pager font".

(see standard font for details)

See also PagFontName.

1.84   Configuration File/Pager

Default: <ENV:Pager, else builtin>

This entry replaces the value read from "ENV:Pager" (if any).

The value "builtin" forces to use the internal pager.

1.85   Configuration File/ReverseSorting

Default: Yes

By default Elm offers you the messages on the index screen in reverse order (newest first, oldest last). If you wish to see the oldest first and newest last (e.g. for mailing lists) set the value to anything else than "Yes".

See also Sorting.

1.86   Configuration File/Shell

Default: newshell

Not everyone uses hotkey utilities like AutoCLI or DMouse. The "shell" command starts an user definable command (need not has to be a shell). The idea is that you make Elm's screen public; so the shell (or any other application) can open it's window on Elm's screen (and not far away on the workbench). The program is run synchronously; use "run" if you want the program to run asynchronously.

This entry replaces the value read from "ENV:Shell" (if any).

1.87   Configuration File/PipeDefault

Default: <none>

The default string for Elm's pipe command.

1.88   Configuration File/SaveDirectory

Default: <current directory>

Default directory for the file requester when saving/writing messages. Once you have changed the directory within the file requester, Elm remembers it and will make that the default directory for further appearances of the file requester. If you cancel the file requester, Elm forgets about your selected path and will use the original default directory instead.

1.89   Configuration File/FilenameOffer

Default: USER

Default file name for the file requester when saving/writing messages.

SUBJECT : subject of the current message (without special chars) FROM : complete "From:" line REPLYTO : complete "Reply-To:" line (useful for mailing-lists) ADDRESS : e-mail address USER : user (account) name NAME : full real name MSGID : message id

1.90 Configuration File/FilenameBuffer

Default: No

Not only remember last directory when saving but also filename.

1.91 Configuration File/TopDownScrolling

Default: Yes

In older versions Elm's internal pager scrolled bottom-up. This is (because of the console.device overhead) very slow on M68000 based machines. For higher scroll speed there is top-down scrolling which doesn't look as good as bottom-up but is much more faster.

Any other than "Yes" enables old-style bottom-up scrolling.

1.92 Configuration File/SigDashes

Default: No

Do we want dashes ("-- \n") above signatures?

(News 2.11 compatibility and convention)

Most AmigaUUCP programs expect the signature identification ("-- \n") within uulib:<user>.signature. This is different from most Unix programs but has become a standard on the Amiga.

1.93 Configuration File/UseOwnScreen

Default: Yes

Rather than opening its own screen you can make Elm opening its window on the workbench (or default public) screen. Under AmigaOS 2.0 (or higher) you can iconify AmigaElm if run in its own window.

1.94 Configuration File/WorkbenchToFront

Default: No

Before invoking the editor you can make Elm calling WBenchToFront() to bring the workbench screen to the front. This might be useful if your editor opens on workbench screen (instead of opening an own screen or appearing on Elm's public screen).

1.95 Configuration File/ElmToFront

Default: Yes

After invoking the editor Elm explicitly brings its screen to the front. If you do not want this feature set value to any other than "Yes".

1.96 Configuration File/IOWindow

Default: CON:20/20/640/200/AmigaElm-IOWindow/CLOSE/WAIT/AUTO

Interactive input/output stream when launching external programs like MetaMail. Read your Amiga manual on how to set characteristics of console I/O windows.

This variable has no effect if you start Elm in terminal mode.

1.97   Configuration File/PubScreenName

Default: AmigaElm

Under AmigaOS 2.0 (or higher) you can make screens public so that other applications can easily open their windows on YOUR screen rather than on the workbench.

Names of public screens must be unique. So Elm appends automatically a sequence number to the name. Thus, "AmigaElm" becomes "AmigaElm.1". If there is already a public screen with this name, Elm increases the sequence number. If this number becomes higher than 999, Elm quits with a warning.

NOTE:

Do NOT use spaces (or tabs) within the name!

Example of using public screen names:

run elm screen=BonzElm newshell con:////HaveFunWithElm/SCREENBonzElm.1

The shell now opens on Elm's screen instead of the default public screen (which is normally the "Workbench"). When you quit Elm you have to close all visitor windows first. But don't panic, Elm will ask you for it ...

1.98   Configuration File/ShanghaiWindows

Default: <current mode>

Under AmigaOS 2.0 (or higher) it's possible that an application with an own screen "hijacks" windows of other applications. To hijack windows your screen must be the default public screen (see DefaultPubScreen).

Shanghai works for all windows which were intended to open on the workbench screen. "Yes", if you want Shanghai.

Note: This is a system global mode!

1.99   Configuration File/PopPubScreen

Default: <current mode>

Under AmigaOS 2.0 (or higher) it's possible that a screen always comes to front if a window is opened on that screen. In my opinion it's very annoying, but one want to activate this mode with "Yes".

Note: This is a system global mode!

1.100   Configuration File/DefaultPubScreen

Default: No

Under AmigaOS 2.0 (or higher) there is always a default public screen. Normally this is the Workbench, but you can make any screen the default public screen. Any application that asks for no special screen will then open on this default public screen. This feature is enabled with "Yes".

1.101   Configuration File/Width

Default: -1

Width (in pixels) of the Elm screen (or workbench window).

Use "-1" for the default screen width (GfxBase->NormalDisplayColumns).

1.102   Configuration File/Height

Default: -1

Height (in pixels) of the Elm screen (or workbench window).

Use "-1" for the default screen height (GfxBase->NormalDisplayRows).

1.103   Configuration File/TopEdge

Default: 0

Top edge offset (in pixels) of the Elm screen (or workbench window).

(automatically sets bottom edge offset to the same value)

1.104   Configuration File/LeftEdge

Default: 0

Left edge offset (in pixels) of the Elm screen (or workbench window). (automatically sets right edge offset to the same value)

1.105   Configuration File/DisplayID

Default: <clone workbench>

With AmigaOS 2.0 there is a new way for setting the display mode of a screen. It's done with a special identifier, the "DisplayID". First you can assign a name (like to ones from the "ScreenMode" preferences program) to the value of this keyword. For example "NTSC:High Res Laced" (don't forget the qoutes if there are spaces in the name). The names are case-sensitive!

If Elm cannot resolve the name it converts the string to a hexdecimal number. In this case the DisplayID must start with "0x" and must be a 32-bit hexadecimal number. Look at your includes <graphics/displayinfo.h> or <graphics/modeid.h> for further information.

If the given name/number is illegal, Elm aborts with a warning.

NOTE:

The display names change from OS version to OS version. AmigaOS 2.1/3.0 uses other names than old release 2.0! (the ID number - of course - stays the same)

1.106   Configuration File/OldLocking

Default: No

AmigaUUCP 1.16D (Matt Dillon) has introduced a new locking mechanism via the OwnDevUnit.library. This mechanism needs no more temporary files in the T: directory, so it's much more smarter and user-friendly (information hiding). Set the value to any other than "Yes" if you want the new locking and if you are absolutely sure that all your other applications are also supporting this locking method.

If you do not have installed the OwnDevUnit.library, Elm has an automatic fallback to the old locking style. Set "OldLocking Yes" to avoid warning messages if you don't have installed OwnDevUnit.library.

It's necessary that all applications use the same locking mechanism. Older versions of AmigaUUCP (like Ingo Feulner's AmigaUUCP Plus or all previous versions from Matt Dillon (<= 1.15)) are still using the old locking method with temporary files.

This keyword/value overrides the "ODULocking" setting from the system global configuration files "s:UUconfig" and "uulib:Config".

1.107 Configuration File/AppendSignature

Default: NoReply

This controls the default for the "Append Signature"-checkbox in AmigaElm's Send-off Window. You may decide to append the signature always or never (by default). But there's another choice:

On replies (Subject begins with "Re: ") there should be no need to include the signature again and again. Your mail partner already knows your name, telephone number and mail accounts. The signature becomes redundant on replies. So, don't waste bytes on the net! With "NoReply" both mail partners send their signature exactly once, further mails in a mail-thread will have no signature appended.

Meaningless how you set this variable, when the Send-off Window appears you have the "last word" if you want to append the signature - or not.

Currently, the following modes are available:

Yes : by default, append signature always (not recommended) No : by default, append signature never NoReply : by default, append signature not on replies (see above)

1.108 Configuration File/MetaMail

Default: Ask

Elm has built-in MIME support for "text/plain" messages. But there are lots of other formats which cannot be handled by Elm's internal pager or standard external pagers (eg, MultiView, More, MuchMore, Most, Less, Leggi). If you wanna be able to display all that bologna, you'll need "MetaMail" (see section MIME).

By default, Elm will ask you if you want to display a MIME message that cannot be handled internally. So you may answer "no" if you don't want it displayed with Metamail or if you don't have Metamail installed.

You can suppress Elm asking you on every unknown MIME message.

You may force Elm to ignore Metamail completely; this way, all unknown MIME messages are displayed literally (eg, if you don't have Metamail installed and don't intend to install it in the near future).

If you like Metamail or get MIME messages very often, Elm can be configured to call Metamail automatically on unknown MIME messages. You will not be asked any more.

Currently, the following modes are available:

Yes : always call MetaMail on unknown MIME messages No : never call MetaMail on unknown MIME messages Ask : ask if MetaMail should be called

1.109 Configuration File/Sorting

Default: Natural

Normally Elm displays the messages on the index screen in natural order (as they reside in the mail folder). You may wish to change the sorting criterion.

Currently, the following modes are available:

NATURAL : original order in mail folder, arrival on system SUBJECT : subject line without leading "Re: " DATE : date, not yet implemented -- use "Natural" instead AUTHOR : sender's real name (From) RECEIVER: receivers's real name (To) LINES : number of lines (message body w/o header) LENGTH : number of bytes (message body w/o header)

See also ReverseSorting.

1.110   Configuration File/PrintCmd

Default: copy % prt: (already implemented? don't think so...hmmm?)

This is very similar to the pipe command except that it is intended to send the current (or tagged) message(s) directly to the printer or a special filter program that reformats the text.

1.111   Configuration File/QuitEmpty

Default: No

By default Elm's screen comes up even if there's no mail for you (that means, the folder exists and has a length of null bytes). If you want Elm to quit automatically if the folder is empty, then set the value to "Yes".

1.112   Configuration File/KeepEmpty

Default: Yes

If (after saving) the incoming mailbox is empty (all messages deleted or stored in the given received folder) Elm may delete the empty (null bytes long) mailbox file. This makes it easier to check for new mails (if there's a file, there must be mail ;-)

If you don't want the empty mailbox file kept (null bytes) then set the value to any other than "Yes".

1.113   Configuration File/ReadContentLength

Default: No (not yet implemented)

Most modern mail systems add a "Content-Length: <#bytes>" header line to the mail header. It gives the total number of bytes of the message body (thus, not including the header). This mostly redundant information is very useful for a noticable faster read-in of the folder because the body part of each message could be totally skipped and left uninterpreted. This tricky method has a minor disadvantage: the signature part of a message cannot be recognized, so there's no way to hide the signature within the built-in pager.

If you enable the feature with "Yes", the keyword "ShowSignature" will be automatically set to "Yes", too.

NOTE:

Currently this setting is used to let Elm compare the calculated length of a message against the value from "Content-Length:" (header). If both values differ Elm prints a warning. At this time "Content-Length:" is not reliable enough to base message separation on it. So use the warnings to make "Content-Length:" more reliable that it can be used in future versions.

1.114   Configuration File/SendContentLength

Default: No

Add a "Content-Length: <#bytes>" to all outgoing mail. This line contains possible useful information for some mail systems. If you don't care about some additional bytes it's recommended to enable this feature with "Yes".

NOTE:

If you're using this feature, you MUST NOT edit any mails within the spool directory without recalculating the correct content length and updating the Content-Length header line.

Mails with incorrect Content-Length may be corrupted!

1.115   Configuration File/MimeCoding

Default: None

Elm supports different content transfer encoding types for outgoing MIME messages. Some modes are also understandable by old mail systems (which cannot handle MIME messages), other modes will need a MIME-compatible mail systems to be decoded correctly.

The following modes are available:

NONE : No coding at all, no MIME specific header.

7BIT : Outgoing mail is guaranteed to be 7-bit-clean. (no "real" umlauts possible, german umlauts are converted (eg, "ä" to "ae"), 8th bit is stripped off if conversion not possible)

8BIT : Full 8 bit (umlauts possible), but text lines must have "normal" length and all links between you and receipient have to 8-bit-clean.

BINARY : Full 8 bit, no limit for line length, but links still have to be 8-bit-clean. (absolutely useless mode in my opinion)

QUOTED-PRINTABLE : Preferred mode for text mails with umlauts, all special chars (ctrl-chars, 8th bit etc.) are escaped. Mostly human readable even with mail systems not supporting MIME.

Recommended to be default for all future mails!

Because of its tricky encoding does not need to have 8-bit-clean links.

BASE64 : Totally encoded, similar to uuencode/uudecode. Text is not human readable and grows in size about 33%. Perfectly suited for sending real binaries, because you don't need the extra uuencode/uudecode step.

Example: Sender types "elm encode=base64 to_adr <bin_file" and receipient only writes this message to a file using Elm's internal w)rite w/o header functions. Decoding is done automatically.

When replying to a message and MimeCoding is unequal to "NONE" then Elm's use of MIME depends on the header of the original message. If the original has a MIME header then MIME is the default for the reply. Tf the original has no special MIME keywords then the default for the reply is no MIME.

1.116   Configuration File/Signature

Default: (1) "<ENV:Signature>" (2) "<$HOME>/.signature" (2) "uulib:<user>.signature" (3) "uulib:.signature"

The filename where Elm should read the signature from.

1.117   Configuration File/CancelledFolder

Default: <don't save cancelled messages>

The filename where Elm should save cancelled messages, if user selects "forget" in send-off menu, for example "uumail:mail.cancelled".

1.118   Configuration File/NoMessageId

Default: No

Elm generates own Message-Ids. This is useful when saving outgoing mail so that it can be referenced at a later time. I've been told that older versions of Dillon's sendmail (until Dec '93) always create a Message-Id line, even if there's already one present in the header.

The only workaround for this bug in Dillon's sendmail is to forbid Elm to create its own Message-Id. Set "NoMessageId" to "Yes" if you don't want Elm's Message-Id.

1.119 AmigaElm/Command Line Options

Command Line Options

Usage: elm F=FOLDER/K,U=USER/K,S=SUBJECT/K,NE=NOENV/S,V=VERBOSE/S, NA=NOARC/K,NZ=NOZERO/S,TO/K,SCR
TERM=TERMINAL/S,MAIL_TO

Usage: elm [-f <folder>] [-u <user>] [-s <subject>] [-scr <name>] [-enc <type>] [-a] [-e] [-v] [-z] [<mail_to>]

Unix style options

$ elm -u postmaster -f dh0:important_mail $ elm amk@zikzak.in-berlin.de $ elm -v -e $ elm -f [file-requester!]

"elm -ve" is not allowed, use "elm -v -e" instead.

Amiga style options

$ elm user=postmaster folder=dh0:important_mail $ elm amk@zikzak.in-berlin.de $ elm verbose noenv $ elm f= \ $ elm folder=
> all equivalent [file-requester!] $ elm folder /

"user=postmaster", "u=postmaster", "user postmaster" and "u postmaster" are all equivalent.

It's allowed to mix Unix and Amiga style options!

Available options

F, FOLDER, -f, -folder <foldername>

Set folder to the following argument. Is no argument given a file requester pops up to ask you which folder do you want. The behaviour of this options changes to previous versions of Amiga Elm, now this options only sets the folder (like Unix Elm) and nothing else. By default (no options -f), Amiga Elm searches in "uumail:" for a folder with the name of your username.

U, USER, -u, -user <username>

Set username. This overrides S:UUConfig, uulib:Config and Env:UserName. If Elm is invoked without a mail-to address and no additional option for a special folder, this options changes the name of the default folder to "uumail:<username>".

NA, NOARC, -a, -arc

Don't archive outgoing mail in SentArchive (though it's editable when sending mail interactively but totally ignored). Useful when sending binaries via redirection (eg, "elm enc=base64 noarc s=binary.lha amk@zikzak <binary.lha" or "uuencode binary.lha binary.lha | elm enc=none noarc s=binary.lha amk@zikzak").

NE, NOENV, -e, -env

Don't read in any environment variables.

NZ, NOZERO, -z

Don't start Elm if folder is empty. (empty means: length of null bytes, NOT non-existant folder!) This overrides QuitEmpty from .elm/elmrc.

SEC, SECURE, -sec, -secure <level>

Start Elm in secure mode. No external programs or shells can be started. Useful in conjunction with terminal mode. <level> is either "1" or "2". Level 1 is for use with MultiUser (muFS), level 2 for full security from within AmiElm without any further ado. Following functions are disabled in secure mode 1 and 2: - 'l' (pipe) - '!' (shell) - '/' (extra programm) - 'c' (change folder) - 'p' (print) - '*' (print) Following functions are disabled in secure mode 2 only: - 's' (save) - 'w' (write) - 'C' (Copy to folder) - 'E' (Export) [removed from code]

V, VERBOSE, -v, -verbose

Be a little bit verbose. Maybe helpful for hunting bugs or to make detailed bug-reports.

TERM, TERMINAL, -t, -term, -terminal

Start Elm in terminal mode. No intuition interface, instead pure input/output (ANSI/VT100) to the console window (stdin/stdout). Also meant for use via AUX: or a similar network environment. Use local environment variables LINES and COLUMNS to specify dimensions greater than 80x20 characters.

S, SUBJECT, -s, -subject <subject>

When sending off new mail (don't forget a mail-to address!) this set's a default subject (which is editable of course).

SCR, SCREEN, -scr, -screen <screen>

Set name of public screen. This overrides PubScreenName from .elm/elmrc.

ENC, ENCODE, -enc, -encode <content-transfer-encoding>

Set mode for content-transfer-encoding. This overrides MimeCoding from .elm/elmrc.

H, HOME, HOMEDIR, -h, -home, -homedir <directory>

Set home directory. Overrides $HOME (if set).

TO <mail-to address>

Special option to explicitly give a mail-to address.

Available modes

There are three modes to start Elm. The first one is without a mail-to address (even not with the "TO" option). This mode lets Elm act as a normal mail reader. Elm will load a folder and let you browse through it.

Examples: "elm" "elm -u root -e"

The second mode is called the "send-only mode". This will allow sending of a single message to the specified recipient with all the options usually available in the Elm system itself. Elm puts you into an editor for you to enter the text of the message.

Examples: "elm amk@zikzak.in-berlin.de" "elm amk@cs.tu-berlin.de verbose"

The third mode is called the "redirection mode" or "non-interactive mode", because Elm will not prompt you for anything. Elm expects at least a mail-to address and a text file on standard input. It's not allowed to redirect from an interactive device/handler like CON:. This mode is very similar to the second mode, except that you cannot edit the contents of your mail. Elm will behave like a user-friendly front-end to sendmail.

Examples: "elm <s:startup-sequence amk@zikzak.in-berlin.de" "uuencode ram:wb21.lha wb21.lha | elm amk@fu-berlin.de"

Due to software limitations it may be possible that the third mode does not act as you might expect. Some (pipe) handlers don't support all the needed functions. Please, start Elm in VERBOSE mode and let me know what happens.

Setting the username

At first Elm scans the command line options for the "USER" (or equivalent) keyword. This keyword beats everything. If this fails, Elm looks for the environment variable "env:UserName". If this variable is unset, Elm searches in S:UUConfig or uulib:Config for a variable called "UserName". Very easy, isn't it?

Setting the folder name

The default name of the folder depends on the username. If no option for a special folder is given, the folder always defaults to "uumail:<username>". You can change the name of the folder by the "FOLDER" (or equivalent) option. If you don't give an argument after the option, a file requester will please you to select a folder file. The starting directory for this requester is "uumail:", but one may change this.

1.120 AmigaElm/Additional Features

.elm/elmheaders

The mailer provides a facility for including customized header lines in the messages you send. If you have an .elm/elmheaders file, the mailer will include its contents immediately after the regular headers of all outbound mail. Each line must be terminated with a linefeed. Here's an example .elm/elmheaders file:

Organization: Department of Fun & Games Cc: bonzo@tic.tac.toe Reply-To: john_doe@tic.tac.toe X-Phase-of-Moon: blubber

These lines will appear after all other header lines in the message.

.elm/smartheader

Sometimes you don't want full-blown headers. You don't want the file .elm/elmheaders included, and you don't want all that redundant header stuff. Of course, you can manually re-edit a message but wouln't it be easier if Elm does this job for you when sending mail to special recipients?

The file .elm/smartheader contains DOS-patterns (one per line). Every line is compared against every recipient (To, Cc, Bcc). If a pattern matches then you get a small, compact header.

.elm/smartsignature

Sometimes you don't want the signature file to be appended. Of course, you can toggle it simply in the "Send-off Window" but wouln't it be easier if Elm does this job for you when sending mail to special recipients?

The file .elm/smartsignature contains DOS-patterns (one per line). Every line is compared against every recipient (To, Cc, Bcc). If a pattern matches then the default in the "Send-off Window" will be to append no signature.

Folder changes

While running, Elm checks the mail folder on every action taken by the user. Elm compares date and length of the folder with previous values and re-reads the folder if it has been changed. This means you can run Elm in background and Elm re-reads the actual folder automatically if new mail comes in.

Folder backup

Elm tries to handle with your data files very safely. Before re-writing a folder to disk, it is backed up to "<filename>.o" (normally located in your mail directory "uumail:"). Therefore it should be very difficult to lose some of your important data during a session.

1.121   **AmigaElm/Aliases**

Aliases

This part of the document is intended as a supplement to the Elm Users Guide and is only of interest to those users desiring more knowledge about how aliases work and how to create strange and exciting aliases for their systems (okay, so maybe it's not that exciting!)

The most simple type of aliases in the Elm system are individual user aliases. These are made up of two parts (separated by an equals sign '='):

aliasname list = address

Where the aliasname list is either a single aliasname or a list of aliasnames separated by commas. Aliasnames can be any combination of letters, numbers, dashes (`-'), periods (`.'), or underscores (`_'). Letter case _is_ significant. That is `FRED' and `Fred' are _not_ identical.

Address is either the user's full electronic mail address or, if the machine routing database is installed, the minimum address needed to specify the destination. For example, say our routing database contained information on how to get to machine `hp-sdd' and I wanted to have an address for my friend Ken there - I could have his address specified as simply `ken@hp-sdd' (or alternatively `hp-sdd!ken' since the two are functionally equivalent).

Let's get on to some examples, shall we?

Consider this excerpt from my own .elm/aliases file:

Mail aliases for Dave Taylor # Friends from UCSD addie = addie@hp-sdd.SDD.HP.COM (Addie Silva) frank,minjarez = Minjarez.Verify@dockmaster.arpa (Frank Minjarez) pizzini = hplabs!ucbvax!ucdavis!pai!ken (Ken Pizzini)

Note that the alias for Frank Minjarez has two aliasnames associated with it, frank and minjarez. Also notice that the first and second aliases use the ARPA Internet style naming convention of user@machine whereas the third uses the UUCP notation of machine!user. For the most part, the notational format is transparent within the alias file - the only time it does make a difference is if you have to specify more than the machine that the user is receiving mail on (as in the third example above). Suffice to say that if you must specify any sort of uucp route that you should use the uucp notation as much as possible to ensure that the system expands the correct machine name. Similarly, if you're bouncing mail through different ARPA Internet sites (to get to a foreign system, as in the example below) you should use the notation that system expects:

listserv%byuadmin.BITNET@rugters.edu

After the confusion of user aliases, group aliases are even more fun! For the most part the notation is very similar:

aliasname list = list of people

Where aliasname list is exactly equivalent to the corresponding fields in user aliases. The interesting part is the list of people field; this field is actually in the same notation as the aliasname list, so it's really quite easy to create. It's best to illustrate by example:

friends, mypals = joe, larry, mary, joanna, nancy, amk@zikzak

Each of the people in the list must be a defined alias (anywhere in the alias file) or a valid mail address. It's allowed to use aliases as well as complex addresses. (this differs from the original Unix Elm!)

The major limitation with aliases is that Elm does not has its own alias editor. Create your aliases with your favourite text editor and save this file to .elm/aliases. Elm reads in the alias list once upon startup. You can change .elm/aliases at any time from within Elm so that Elm accepts the changes immediately. A running Elm does not detect changes in the alias file if they were made in the background by another program.

If you have defined an alias twice Elm ignores the second one and prints out a message.

The maximum line length is 1024 characters. Up to 20 levels of aliases (nested aliases) are allowed. For example, a nested alias:

dummy1 = user1, user2, user3 dummy2 = user4, user5, user6 myalias = dummy1, dummy2

1.122   AmigaElm/MIME

MIME -- Multipurpose Internet Mail Extensions

no documentation yet... read rfc1521 for details :-)

Please, have a look at the "ReadMe.Txt".

If somebody wants to write a small "Guide to AmigaElm, MIME and Metamail", I'll include it here...

Very short introduction

Elm has built-in MIME support for "text/plain" messages. But there are lots of other formats. If you wanna be able to display all that bologna, you'll need "MetaMail" (current version for Amiga is 2.3a). Metamail is not only a single program; it's a package with documentation, binaries and all that stuff.

Use the variable MetaMail in your .elm/elmrc file to control AmigaElm's handling of Metamail.

Metamail is available on "ftp.cs.tu-berlin.de" (Internet FTP) and via normal e-mail from "mail-server@cs.tu-berlin.de".

If you're not familiar with our mail-server try the following:

1. mail to "mail-server@cs.tu-berlin.de" 2. write "send /pub/amiga/uucp/metamail-2.3a.lha" in the mail body 3. write "send /pub/doc/rfc/rfc1521.Z" in the mail body (or "send /pub/doc/rfc/rfc1521.ps.Z") 4. wait some days ;-)

Really easy, isn't it?

Some tips for metamail

I don't have most of the programs named in metamail's "mailcap" file (which resides in uulib:). So far I only needed "text/plain" (for base64 coded mails) and "text/richtext". My favourite entries are:

text/plain; multiview pubscreen=`getpubscreen` %s text/richtext; showRichtext %s

Where "type `which showRichtext`" looks like:

.KEY Richtext/A .BRA { .KET }

richtext > {Richtext}.txt -f {Richtext} multiview pubscreen=`getpubscreen` {Richtext}.txt delete {Richtext}.txt QUIET

What the hell is "getpubscreen" you may ask. GetPubScreen is a small utility (initially by Michael 'Mick' Hohmann) which prints the name of the front-most public screen. So all metamail-related window will open on Elm's public screen. What? You don't have AmigaOS 2.0 (or higher)? Sorry, you loose ;-)

1.123 AmigaElm/Security

Security: Encryption, Decryption and Signing

AmigaElm has the ability to encrypt and decrypt messages. AmigaElm is totally configurable in the way messages are encrypted and decrypted. The various encryption modes (encryption, digital signature etc.) of outgoing messages are controlled via the Send-off Window. The default encryption mode is set with CryptMode in your .elm/elmrc file. Decryption from within Elm is done via the appropriate command key for decryption, Decrypt key.

Two separate programs of your choice do the actual encryption/decryption of the messages.

See also Decrypt (decryption of incoming mail) and Encrypt (encryption/signing of outgoing mail) in .elm/elmrc.

PGP -- Pretty Good Privacy

PGP is a program that gives your electronic mail something that it otherwise doesn't have: Privacy. It does this by scrambling or encrypting the message so that nobody but the intended receiver can read it. When encrypted, the message just looks like a meaningless jumble of random characters. PGP has proven itself quite capable of resisting even the most sophisticated forms of analysis aimed at reading the encrypted text.

PGP (ported to Amiga by Peter Simons) is very popular for crypting messages. Included in this package is "PGPMore" by Peter Simons (simons@peti.GUN.de) to decrypt PGP-encrypted messages. You need at least "PGPAmiga 2.3a.3" for use with the included scripts.

If you have lots of encrypted mails in your folder (and decrypting and replying each of them separately is quite uncomfortable), then you may decrypt your complete folder at a single blow. The included script "DecryptFolder" shows you how. It doesn't matter if there are no encrypted messages in the folder or if encrypted and normal messages are mixed.

Please be careful with deciding to do that, because the e-mail will be stored in plain text on your hard-disk, accessible to anyone. You give up a lot of security if you do so!

Also available from Peter Simons is "PGPSendmail" (2.0), a front-end for sendmail, that automatically encrypts all messages to specified recipients.

AmigaPGP and PGPSendmail are available from any AmiNet site (in util/crypt).

Remark: The current version of "PGPMore" needs your password in the environment variable "$PGPPASS". This may change in future (to a more intelligent password handling).

1.124 AmigaElm/Internet, TCP/IP

AmigaElm with AmiTCP, AS225 etc.

It's true that AmigaElm is designed for use with so called UUCP packages. But AmigaElm doesn't really require UUCP. However, installation is lots easier if you've UUCP already running on your system.

By default, AmigaElm uses uulib:Config (a global UUCP config file) which can be replaced by S:UUConfig. Either this file (S:UUConfig or uulib:Config) must exist and contain entries for "UserName", "RealName", "NodeName" and "DomaiName". OR you provide AmigaElm with the mandatory settings for your name and host via environment variables (RealName, Host-Name, DomainName).

AmigaElm will not access uumail: (this is where UUCP packages store mails) to load mailfolders if you specify the mailfolder file directly with the "-f" option. AmigaElm does not need uulib:.elm/elmrc (etc.) if you set the env-variable HOME (then $HOME/.elm/elmrc (etc.) will be read).

In AmigaElm's config file ".elm/elmrc" (see above) the keywords for SendMail and RMail should be configured to call, for example, "SMTPpost" from Michael B. Smith's "INetUtils" (or similar programs that do mail handling on Internet, eg AmigaSMail by Ralph-Thomas Aussem).

The Sendmail/Rmail is for outgoing mail. If you aren't 24 hours online a day on the Internet you should queue your outgoing mail until you actually have connection to the Internet. Run the SMTPd (eg, from INetUtils) as part of your "startnet" script (eg, AmiTCP) and it periodically checks your mail-spool directory and sends anything it finds.

To get incoming mail you should install something like "AmiPOP". When you run it from your "startnet" script (eg, AmiTCP) it can be configured to periodically check your mailbox while you are online. If new mail arrives, it's transferred automatically to your local mail-folder and can be read with AmigaElm.

To receive mails directly addressed to your local site you should install the SMTPd and configure it in the "inetd.conf" file of your TCP/IP package (mail, port 25).

Dynamic IP addresses

If your IP provider assigns a dynamic address (IP number, host name) to your site on every call, you cannot use it as return address for mails. AmigaElm builds the "From:" line in the mail header automatically, so you cannot change this.

But with the ".elm/elmheaders" file you can set a separate reply address. For example, "Reply-To: john_doe@cool.org" forces all replies to this mail to go to the given address - software must ignore the "From:" line if "Reply-To:" exists. So, it doesn't matter if "From:" is incorrect, "Reply-To:" will fix it.

Don't try to "fix" this problem by leaving out the dynamic host name!

For example, your host name is "slip42.prefect.com". Don't set hostname (nodename, sitename) = "prefect" and domainname = ".com". This confuses SMTPpost/sendmail, because it's now impossible to determine if "user@prefect.com" is a local address on your system or the address on another system that has a dynamic IP connection (like you) to the server "prefect.com".

1.125   AmigaElm/How to install Elm

Installing

Copy the Elm binary to your DOS search path (e.g. "copy elm uucp:c clone"), create a directory "uulib:.elm" and a configuration file "elmrc" in it. It stands to reason that you must have all the assigns and configs you need for AmigaUUCP in general.

Load "uulib:.elm/elmrc" into your favourite editor. You must have at least two entries called SendMail and RMail. If you have no "RealName" in uulib:Config nor a proper initialized "Env:RealName" you must set FullName also.

Aah, and don't forget: Elm is pure ('p' flag) and can be made resident via the "resident" command.

WARNING!

Please adjust the uulib:.elm/elmrc to your personal needs and settings. The files in the uulib/ directory are only EXAMPLES (with MY settings). DO NOT USE THEM. NEVER USE! CAPITO?

Now you can start and use Elm from everywhere. But you may want more informations about advanced usage. Okay, here's some more information...

Command line options always beat their counterparts from the environment ("ENV:" directory) and from the configuration file (by default "uulib:.elm/elmrc"). Elm's configuration file always beats the environment, that's the standard behaviour for all UNIX applications. So it will be a good idea to leave all unnecessary stuff out of the config file and to set the appropriate variables in the environment or the system global UUCP config file (if existing "s:UUconfig", else "uulib:Config"). The environment always beats this system global config file. So we have the following priority order:

Command line options (user=test etc.) Elm's configuration file (e.g. uulib:.elm/elmrc) Local environment (shell) Global environment (env:) UUCP configuration file (e.g. uulib:Config)

Elm also supports multi-user. That is different mail folders, different configuration files and, and, and...

Elm supports the following environment variables:

PAGER : replacement for internal pager SHELL : program to run for "\" command key (see also command key \) PRINTER : printer command SIGNATURE : name of signature file (see also Signature in .elm/elmrc) NOMETAMAIL : if set, no fallback to "metamail" in internal pager EDITOR : favourite editor (see also Editor in .elm/elmrc) AUXEDITOR : editor when in terminal mode (see also AuxEditor in .elm/elmrc) REALNAME : real name HOST : name of site, without domain DOMAIN : domain name (with leading '.') USER : user name (account) LINES : number of lines if in terminal mode (only local env) COLUMNS : number of columns if in terminal mode (only local env) HOME : user's home directory where Elm searches first for .elm/elmrc, .elm/elmheaders, .elm/aliases, .signature etc. HOME_ELM : overrides HOME if present

If HOST, DOMAIN or USER do not exist, Elm will try each of the variables again with an appended "NAME" (HOSTNAME, DOMAINNAME, USERNAME).

1.126 AmigaElm/Useful extra programs

This is description of some useful external utilities. They reside in the "c" directory of the AmigaElm distribution. You should copy them somewhere to your DOS search path so that AmigaElm will find them.

debase64

Decodes plain "base64" text on stdin to 8-bit data (ASCII text or binary) on stdout.

DecodeNeXTmail

Decodes NeXTmail message and displays all its contents on AmigaElm's screen. Needs some UNIX programs like "uudecode", "compress" and "tar".

DecryptFolder

Decrypts a complete PGP-crypted folder.

(use this script only if you are absolutely sure that nobody else can read the decrypted folder)

DecryptMsg

Decrypts a PGP-crypted message and displays it on AmigaElm's screen.

enbase64

Encodes 8-bit data (ASCII text or binary) on stdin to plain "base64" text on stdout.

EncryptMsg

Encrypts a message with PGP (very flexible). Needs PGP.

getactscreen

Get name of active (public) screen.

(needs not to be the front-most or even a visible screen)

getpubscreen

Get name of front-most (public) screen. (screen needs not to be active)

PGPMore

Decrypts a PGP-crypted message and displays it on stdout. Needed by some other programs.

RequestString

Prompts user for a string. Needed by some other programs.

splitheadbody

Splits a mail message on stdin into to two different files: one containing the header, the other containing the message body.

xdel

Recursively deletes files and directories. Does _not_ crash Commodore's RAM-Disk (like some other programs do). Delete-protected files are unprotected first.

This program has no security checks and doesn't ask for confirmation first. Use with care!

xuudecode

Decodes an uuencoded file and saves output to a given directory.

1.127 AmigaElm/Bugs, ToDos, Infos, Thanks ...

Reports

Send detailed bug reports, ideas, flames etc. via e-mail to "elm-fan@zikzak.in-berlin.de" or "elm-fan@zikza.snafu.de" (UUCP/Internet). I need your Elm version number and a reproducable bug generation example.

If you report Enforcer hits please send the complete Enforcer and SegTracker output. Otherwise it's of no earthly use.

Plagiarism

Some parts of this document are taken from the original Elm documentation written by Dave Taylor (taylor@hplabs.hp.com). (my English is too lousy to write all parts by myself :-)

My personal to-do list

- better handling of included mail-headers - more gadtools support - better printer support - ReadContentLength in .elm/elmrc

Thanks

- Kai 'wusel' Siering (best beta-testing, best ideas) - Michael-Wolfgang Hohmann (ideas & funny telephone calls) - Ralph Babel (ideas, tips & tricks, infos) - Holger Lubitz (ideas, infos) - Henning Schmiedehausen (ideas) - Walter Mildenberger (ideas) - Roland Bless (ideas, sources) - Stefan Becker (best flames) - Peter Simons (PGP, PGP, PGP) - Michael B. Smith (AmigaUUCP) - Matt Dillon (AmigaUUCP D) - Ingo Feulner (AmigaUUCP Plus) - all registered users - all the other people who send me bug reports, ideas etc. - all I forgot ;-)

Movie

"Heathers" (Lethal Attraction), best film ever made.

1.128 AmigaElm/Disclaimer

Amiga-Elm is © Copyright 1991-1995 by Andreas M. Kirchwitz, Germany. All rights reserved.

Any possesor of this version of "Amiga-Elm" for the Amiga is hereby granted a non-exclusive license permitting its use and/or redistribution, subject to the following terms and conditions.

Permission is hereby granted to freely redistribute this version of "Amiga-Elm" via electronic bulletin board systems (BBS's), freely redistributable disk collections (such as provided by Fred Fish), service bureaus (BiX, GEnie, CompuServe, etc), and networks such as USENET, BITNET, and Internet, provided that such distribution includes this unmodified License, and all of the documentation files (README and .doc files), in addition to the executable, and with all copyright notices intact. Access to the source code must also be available and allowed.

This archive may be freely redistributed, but only in totally unchanged state, i.e. no files can be added, deleted, modified etc. All copyright notices in the program and its documentation must remain on their places. Also ".displayme" and other files, usually with "wonderful" ANSI graphics, so obvious at various BBS's, cannot be added.

1.129   AmigaElm/Author Info

Snail mail:

Andreas M. Kirchwitz Seesener Str. 69 D-10709 Berlin Federal Republic of Germany

Phone (voice): +49 (0)30 86409100

UUCP/Internet:

elm-fan@zikzak.in-berlin.de elm-fan@zikzak.snafu.de

Internet Relay Chat (IRC) nick:

bonzo

Mailbox/BBS (EazyBBS): (HST, V.32, V.32bis)

ZikZak Inc., +49 30 8623376, 15-21h, ID: Big Bonzo (Sysop) Leguan's Byte Channel, +49 30 8122442, 24h, ID: Big Bonzo