```
####    ####               ###         ######     ####    ####  ####    ####
#####   ####               ###       ##########   #####   ####  ####    ####
######  ####              ####      ####    ####  ######  ####  ####    ####
####### ####   ######   #########   ####          ####### ####  ####    ####
############ ########## #########   #### #######  ############  ####    ####
############ ###    ###   ####      #### #######  ############  ####    ####
#### ####### ###    ###   ####      ####    ####  #### #######  ####    ####
####  ###### ###    ###   ####      ####    ####  ####  ######  ####    ####
####   ##### ##########   #########  ##########   ####   #####   ##########
####    ####   ######       ######     ######     ####    ####     ######
```

Version 1.6
March 25, 1994

REFERENCE MANUAL

```
                      %%%%%%%%%%%%
                  %%%%############%%%%
%%%%%%          %%####  ########  ####%%          %%%%%%
%%##%%%%      %%####  ############  ####%%      %%%%##%%
%%%%%%      %%####    ###   ##  ####%%      %%%%%%
  %%##%%  %%##########  OO  ##  OO  ########%%  %%##%%
  %%####%%############      ##      ##########%%####%%
    %%%%%%#####    #####    ####    ###    ###%%%%%%
            %%######  ##  ##################  ##  ####%%
            %%###########                     ########%%
            %%##############     XXXXXX     ##########%%
            %%##################          ############%%
            %%########################################%%
            %%########################################%%
            %%########################################%%
            %%########################################%%
             %%######################################%%
              %%####################################%%
               %%##################################%%
                %%################################%%
                  %%############################%%
                    %%########################%%
                      %%%%%%############%%%%%%
                      %%####%%%%%%%%%%%%####%%
                      %%####%%        %%####%%
                    %%%%%%%%%%        %%%%%%%%%%
                  %%########%%        %%########%%
                  %%%%%%%%%%%%        %%%%%%%%%%%%
```

TABLE OF CONTENTS

1. INTRODUCTION

NotGNU - This is not the GNU Emacs Editor. A text editor similar to GNU Emacs but much smaller and lighter. Compare a (quick) economy car to a mobile home and that's NotGNU vs GNU emacs. They'll both get you where you want to be, with more or less gadgets along the way.

The editor was based on MG2a, which was based on the original Conroy emacs. Some of the basics remain the same, but I have heavily modified the code to support the windowing environments, support large files using Windows virtual memory, and to make it's basic editing look more like GNU, or to add whatever I needed. Your mileage may vary.

I have built it a number of different Windows, Windows NT/Intel, and Unix machines; others have also successfully compiling it for e.g. Linux and NT/AXP. Source code for Unix and X11 builds is available, see details later in the manual.

> This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

I reserve all rights to this software, both in source and binary form. You may freely redistribute it, as long as you do not make any profit from it, or include it in a product sold for profit. If you make modifications and/or bug fixes to the source, please feel obliged to send them to me. I will maintain and support versions I distribute as time and money allows.

Please send me mail if try this editor; I would appreciate any/all comments. Otherwise, enjoy and find it useful; that's the overall goal. Donations graciously accepted.

Julie Melbin, E-mail: <julie@world.std.com>
P.O.Box 1007,
Groton MA 01450

———

This document was created by Tor Slettnes on August 9, 1993; last update was for NotGNU version 1.6 on March 25, 1994. Comments, suggestions and questions are welcome; please send e-mail to <tor@netcom.com>.

2. CONFIGURING NOTGNU

After installing NotGNU in a directory of your choice, it should be possible to run the editor by typing the name of the executable or double-clicking on the file name (Windows 3.x/NT). NotGNU is distributed with a sample start-up file (NOTGNU.ST) that provides some basic default settings as well as some helpful hints on how to customize these. The following section should provide the information required to customize NotGNU further, to suit your individual preferences.

2.1. Window settings: Size, Fonts, Colors, MenuBar

[Windows versions only]

If you click once on the Windows Application Control Menu in the upper left hand corner of the NotGNU window, you will see that there are some NotGNU specific items in addition to the standard Windows ones. Use these to toggle NotGNU's menu bar on and off, select its font and colors, and to save current settings. "Save Settings" will save current position and size of the window, and disable the "Welcome!" message box from appearing every time you start NotGNU.

The settings will be saved to the file NOTGNU.INI in your Windows directory.

2.2. Using alternative command interpreters

If you want to use alternative DOS command shells, such as 4dos or Ndos, for commands like compile, shell-command, and shell-window, you need to do two things:

- Specify the MESHELL and MEEXEC environment variables.
- Configure the corresponding PIF file.

The following two sections will explain in further detail how to go about this.

2.3. Environment Variables

The following environment variables will be interpreted by NotGNU, if present: HOME, MEHOME, MEMAKE, MESHELL, MEEXEC.

* HOME should point to the directory of your NOTGNU.ST or .NOTGNU
  file, and if present, the notgnu.alr file. Also, NotGNU temporary
  files, such as MExxxxx.LOG and MYMACS.MSG, are stored here.
  Default: The directory of your executable.

* MEHOME, if present, supersedes HOME.

* MEMAKE contains the default command to suggest for compilation.
  Default: "nmake" under DOS and Windows, and "make" under UNIX.

* MESHELL contains the path to the command interpreter or shell
  for the Windows versions. DNOT uses %COMSPEC% instead.
  Default: "command.com"

* MELOG contains the name of the log file. Default is meXXXXX.log,
  where XXXXXX is a unique processid (or like).

* MEEXEC contains a format string used for spawning background
  processes such as compilation in the Windows versions. Default
  formats are:
```
    Windows NT    "%s /c %s 1>%s 2>&1"
    Windows 3.x   "%s /c %s > %s"
```

  Each %s is substituted with, in order of appearance:
    - The command interpreter to use (MESHELL)
    - The compile command given by the user in the minibuffer
    - The output log file name (MELOG)

If you are using 4dos or Ndos, you should insert into your 4start.btm/Nstart.btm or autoexec.bat:
```
  set MESHELL=%COMSPEC%
  set MEEXEC=`%s /c %s >& %s`
```
(This will redirect both stdout and stderr to the log file).

2.4. PIF file settings

[Windows versions only]

MESHELL.PIF and COMMAND.PIF are sample Windows Program Information Files that NotGNU uses, in the absence of a MESHELL environment variable, to execute shell-window and shell-command. They are similar to _DEFAULT.PIF, but with increased background priority, execution in background enabled, exclusive execution enabled, and close window on exit.

It is important that you copy COMMAND.PIF to your Windows directory or the directory of COMMAND.COM if you want background tasks like 'compile' to actually RUN in the background under Windows 3.x. Failure to do so means that background jobs will run VERY VERY SLOWLY. Also if your PIF file does not define "Close Window on Exit", then NotGNU can not tell when the spawned task is complete.

If you have specified the MESHELL environment variable, MESHELL.PIF and COMMAND.PIF will not be used. Instead, execution parameters for BOTH shell-command and shell-window will be read from the PIF file that corresponds to the specification in this variable. Suppose, for instance, that %MESHELL% points to C:\DOS\4DOS\4DOS.COM. You would then have to edit the file C:\DOS\4DOS\4DOS.PIF, so that the following are specified (Use the Windows PIF editor):

```
Advanced->Background priority: 90 or more
 Execution in background      : Yes
 Execution exclusive         : Yes
 Close window on exit         : Yes
```

Also, make sure that NO start-up directory is specified; otherwise make/compile will not work properly.

(These are things you should do anyway, even without NotGNU).

2.5. Command Line Switches

The following switches can be given as arguments on the command line when starting NotGNU. Non-Un*x versions accept either '/' or '-' as the switch character (ie. /x and -x for the DOS and Windows versions are equivalent).

If an argument to a switch consists of more than one word, UNIX shells and 4DOS will recognize double quotes (") to enclose it. COMMAND.COM does not do this; however '_' (underscore) can be substituted for a space, so that 'wnot -f set-default-mode_fill' is equal to (set-default-mode 'fill) in NOTGNU.ST.

Available switches are:

-x <function>   Causes <function> to be executed after the file(s) given as arguments on the command line are loaded. For instance, to open a file in read-only mode, use the command:

```
wnot.exe -x toggle-read-only <filename>
```

This mode will be toggled for the first file only.

-f <function>   Causes <function> to be executed as soon as NotGNU is started, before any buffers are loaded. Similar to -f under GNU emacs. For instance, if you want to use DNOT as a text editor, start it with the following command:

```
dnot.exe -f set-default-mode_fill [filename]
```

-p <filename>   [Windows 3.x only] Prints file, then exits. Used for Notepad compatibility. This switch will also work with NotBoot.

-v <filename>   View file; do not allow any editing. This switch will also work with NotBoot.

```
-fg <color>     [X11 only] Text color
-display <name> [X11 only] Display name
-bg <color>     [X11 only] Background/Window color
-font <font>    [X11 only] Font
-fg2 <color>    [X11 only] Touched text color
-fg3 <color>    [X11 only] Echoline and highlight color
-help           [X11 only] Dumps a message
```

2.6. Start-up file: NOTGNU.ST

NOTGNU.ST (or .notgnu under Unix) will basically contain commands that you want to be executed at start-up. Most commands that you can execute at run-time, you can also put into this file.

2.6.1. Basic syntax

The basic syntax is similar to GNU Emacs. The commands are enclosed in a pair of parentheses, '(' and ')':

```
(toggle-just-write-newline)
```

2.6.2. Arguments

Any arguments should be preceded by the ' character:

```
(set-window-size '43,80)
```

2.6.3. OS dependent configuration

For operating system dependent configuration, use:

```
if OS (command)    ; Execute command if running under OS
if !OS (command)   ; Execute command if not running under OS
```

where OS is one of: DOS, Win3, WinNT, Unix, X11 and OS/2. (For the latter, see Section 6, OS/2 Bugs). Note that the syntax is case sensitive. One example:

```
if !DOS (set-stop-enable '0) ; Execute if not running DNOT.
```

2.6.4. Comments

Comments in NOTGNU.ST should be prefixed by a ; (semi-colon).

2.6.5. global-set-key

It is also very common to specify keyboard commands in NOTGNU.ST. Use the following syntax:

```
(global-set-key "\^x!"    'shell-command)          ;C-x !
(global-set-key "\e[A"    'previous-line)          ;ESC [ x
(global-set-key "\efx"    'save-buffers-kill-emacs) ;Alt-F x
(global-set-key "\Insert" 'set-mark-command)       ;Insert key
(global-set-key "\e\Backspace" 'backward-kill-word) ;Esc BackSp
```

Note that on a PC, Alt-<key> and ESC <key> are equivalent. This is often refered to as the Meta key, for historical reasons and compability with some Unix machine keyboards.

The following are legal key names on a PC:

Gray keypad	\Insert, \Delete,n
	\Home, \End, \Page-Up, \Page-Down,
	\Up-Arrow, \Down-Arrow,
	\Left-Arrow, \Right-Arrow
Ctrl + Gray	\Control-Home, \Ctrl-End,
	\Control-PgUp, \Ctrl-PgDn,
	\Control-Up, \Ctrl-Down,
	\Control-Left, \Ctrl-Right,
	\CBackspace, \CDelete
Shift + Gray	\SBackspace, \SDelete
Function keys	\F<n>
Shift + F-keys	\FS<n>
Ctrl + F-keys	\FC<n>
Shift + Ctrl + F-keys	\FCS<n>
Alt + F-keys	\FM<n>
Keypad characters	\Kp-+, \Kp--, \Kp-*, \Kp-/, \Kp-.,
	\Kp-0 .. \Kp-9 [only w/NumLock on]
Characters	<c>
Ctrl + Character	\^<c>
Alt + Character	\e<c>
Esc	\e
Tab	\^i

<n> is 1-10 for the DOS version, and 1-12 for Windows 3 and NT.
<c> is a character

3. EXTRA FEATURES

3.1. NotAlarm

Notalarm is a simple reminder function that can be set via the `alarm' command in NotGNU. Or, if you prefer, you can edit the alarm file (NOTGNU.ALR) directly. Make sure you put NOTALARM.EXE where you put the editor exe file. The NOTGNU.ALR file is loaded from HOME on start-up (see section 3.2, Environment Variables).

Five parameters are expected for each alarm. These are, in order:
Day: 1-31, Month: 1-12, Hour: 0-23, Minute: 0-59, Message: string.
`99' can be used as wildcard (every day, every month, every hour..).
This code has minimal error checking.

3.2. NotBoot

NotBoot is a boot manager for the Windows 3.x and NT versions of NotGNU. Only one instance of NotGNU can run at once on Windows 3.1, but starting up NOTBOOT.EXE with an argument will import the filename given in the argument into the existing NotGNU window; or if none exists, NotBoot will start a new NotGNU.

By associating the appropriate file extensions in the File Manager (or Norton Desktop etc.) with NOTBOOT.EXE, you will be able to edit the corresponding files by double-click on the file icons, even if NotGNU is already running. Note that the HOME environment variable must be set for this to work correctly. A small file called MYMACS.MSG will be created in the directory specified by HOME. Also, if you are using Norton Desktop for Windows, you should make NOTBOOT.EXE your default Windows editor rather than WNOT.EXE.

NotBoot now accepts the following command line switches:

```
/p  Print file, then exit.
    Syntax: notboot /p [filename]
/v  View file (i.e. do not allow any editing).
    Syntax: notboot /v [filename]
```

4. NOTGNU OPERATIONS

4.1. Mouse Operations

The mouse is now supported in all versions except Unix curses.
The DOS version DNOT does not have an external clipboard available,
so all clipboard commands apply to the internal kill buffer instead.

Set cursor position: Left button down-up at desired position.

Resize buffer window: Left button down-drag on modeline that separates the buffers (when multiple buffers are displayed).

delete-other-windows (windowed versions only): Double click left button on modeline of buffer you wish to keep (same as C-x 1 while window you wish to be full screen is active).

Cut or Copy region to clipboard: Left button down-drag over text to be cut or copied. Mark will be set where dragging started, cursor where it stops. DOS version uses the kill buffer.

Paste from clipboard: Right button down-up at desired position. See Section ###, toggle-mouse-paste-where.

Insert file from drag & drop client: Drag file icon from external application, release button at desired position in NotGNU. (Windows only).

Open new buffer w/file from drag & drop client: Drag file icon from external application, release button on the modeline, or NotGNU icon if minimized. (Windows only).

Select file from `dired' or buffer from Buffer List: Left button double-click on filename. (Not available in DOS).

Scrolling: Left button down-drag; move mouse pointer above the top or below the bottom of the screen pane. Wiggling the mouse will slow the scrolling.

4.2. Killing, Yanking, Cutting, Pasting

The windowed versions of NotGNU operate on three different kill-buffers/clipboards for cutting and pasting purposes:

* The internal kill-buffer, with corresponding kill ring, consulted on the following commands:
    - kill-word, kill-line, kill-paragraph, kill-region
    - copy-region-as-kill
    - yank, yank-pop

* A separate buffer for rectangle operations. Use the commands described in section 4.3: Rectangle Operations.

* The Windows clipboard or X11 cutbuffer 0, consulted on the following commands (DOS version uses internal kill buffer):
    - kill-region-to-clipboard
    - copy-region-to-clipboard
    - yank-from-clipboard.
    - (Ctrl-) left button down-drag; (Ctrl-) right mouse button.

4.3. Rectangle operations

NotGNU provides GNU Emacs 19-like rectangle operations. Select a region using the mouse, or set-mark-command followed by a cursor movement; then type one of the following commands:

```
M-x copy-rectangle   ; Copy rectangle between mark and cursor
M-x delete-rectangle ; Erase rectangle between mark and cursor
M-x kill-rectangle   ; Cut rectangle between mark and cursor
M-x clear-rectangle  ; Replace text in rectangle with blanks
M-x open-rectangle   ; Insert blanks into the rectangle (*)
```

To yank back the rectangle buffer into the text, use

```
M-x yank-rectangle   ; Insert rectangle buffer into text (*)
```

(*) TAB characters cause incorrect results in open-rectangle. Also, yank-rectangle will not change the contents of a rectangle to account for any yanking to a different tab offset. Use tabify and untabify as a work around.

4.4. Autocompletion

Pretty straightforward; simply press <Space> or <Tab> to complete a filename or a command while typing in the minibuffer. For filenames, <Space> will list all files, while <Tab> will list only directories (useful for quick access to subdirectories in cases where the parent directory contains many files, such as your windows directory).

'M-x set-gnu' alters the behavior of file completion such that <Tab> attempts full completion without generating a full dired-completion buffer, thus you lose the step-down-into-directory feature defaulted by notGNU. <Space> attempts to complete only to token (.,-) when filling in a file name, a'la GNU. It's still not exactly like GNU, but it's closer.

Wildcards can be used with autocompletion. Use "*.TXT <Space>" to list all files with extension `.TXT'. NotGNU can not open a file with wildcards in the name (i.e. FO*.BAR will not get you FOO.BAR).

4.5. Minibuffer

The minibuffer now accepts the following commands:

C-h, Backpace	Delete character to the left of cursor
C-d, Delete	Delete character at cursor
C-w	Delete word to the left of cursor
C-u	Delete to start of line
C-k	Delete to end of line
C-b, Left arrow	Move cursor left
C-f, Right arrow	Move cursor right
C-a, Home	Move to start of buffer
C-e, End	Move to end of buffer

4.6. Buffer List

Invoked by M-x list-buffers (C-x C-b). The following commands are available in the Buffer List:

u = Use (switch to) this buffer
s = Save this buffer
k = Kill this buffer

4.7. dired

Invoked by M-x dired (C-x d). The following commands are available in "dired":

r = refresh buffer
e = edit this file
f = find-file
v = view-file
d = delete this file. Cannot be undone.
m = mark for batch processing

4.8. query-replace, replace-string, set-case

M-x query-replace (M-%) performs interactive string replacement with user confirmation for each entry, while M-x replace-string performs an unconditional replace.

The following options are available when prompted whether to replace an occurrence with query-replace:

! = unconditionally replace the rest
. = replace this occurrence, and then skip the rest
n, DEL = do not replace this occurrence
y, SP = do replace this occurrence
^G, ESC = quit 'query-replace' mode.

After being done with one buffer, you have the option of moving to the next buffer, skipping the next, or quitting.

4.9. Undo

The way undo works is a bit different from GNU emacs. In GNU, an undo operation is itself pushed onto the undo stack. If you do several undos in a row, it will go back in the stack, but if you do another operation in between two undos, it will start over again from the top (and undo your undos).

NotGNU has a simpler approach; any undo will simply pop out the last executed operation from the undo stack, and the stack will shrink. Undo also takes an argument; 'C-u 5 M-x undo' will do 5 undos. Undo's are stored on a per buffer basis and edits/undos in one buffer do not effect the state in any other.

You can set the maximum depth of the undo stack to some other value than the default 100; use M-x undo-high-threshold. Increasing the depth increases memory usage, setting the size to 0 disables undo. Saving the file clears the undo stack, you can also clear it by hand with M-x empty-undo-stacks to free some memory. The amount of memory required for each undo will vary with the complexity of the editing changes you make. For instance, if you upcase a big region, the whole region was saved in its old state.

4.10. Pulldown and Popup menus

Pulldown menus can be toggled on and off with the "Window Menu" item in the application control menu in the upper left corner or the NotGNU window. Once enabled, items in it can be selected by pulling the menus down with the left mouse button.

A popup menu is now also available; by default bound to the right mouse button. You can add and delete items from this menu; for appropriate commands do a ``M-x apropos RET popup RET''; followed by ``M-x describe-function'' for more information on each command.

Use ``M-x set-popup-button'' to select whether the popup menu should be on the left or right mouse button (Default: right). The original function on this button will then be on Ctrl + Button.

5. BASIC EMACS COMMAND REFERENCE

This is an overview of some of the more common commands available in NotGNU. If you are new to emacs, it may be a good idea to get a hardcopy of the following pages, and keep them next to your workstation.

Complete function descriptions are available via M-x describe-command, and in the file NOTGNU.FUN. This file is also formatted for printing.

First a word on notation:

```
   C-h = Control-H. Hold down the Control key, and press H.
   M-h = Meta-H. Some computers do indeed have a Meta key, others may
         label it differently. Look for "Alt", "Extended Char", etc.
         If your computer does not have such a key, you can use Esc.
         To obtain M-h, you would press and release Esc, then press H.
         On a PC, M-h means Alt + H.

   NG  = NotGNU specific function, not defined in GNU emacs.
   HW  = Hardwired (non-bindable) key, cannot be changed.
   HK  = Hardwired key command, not available as named command via M-x.

   ------------------------------------------------------------------
                       |
   C-b, C-f, C-p, C-n  | backward-char, forward-char,
   or arrow keys       | previous-line, next-line
                       |    Moves cursor left, right, up, and down.
                       |
   M-b, M-f            | backward-word, forward-word
   C-Left, C-right (NG)|    Moves the cursor one word backward or
                       |    forward.
                       |
   C-a,  C-e           | beginning-of-line, end-of-line
   or Home, End        |    Moves to the beginning or end of line.
                       |
   C-v, M-v            | scroll-up, scroll-down
   or PgDn, PgUp       |    Jumps one screen full down or up
                       |
   C-d                 | delete-char
   Backspace           | delete-backward-char
                       |    Delete one character
                       |
   C-Up        (HW, NG)| scroll-one-line-down
   C-Down      (HW, NG)| scroll-one-line-up
                       |
   C-Tab       (HW, NG)| hard-tab
                       |    Insert a <TAB> character
                       |
   Tab             (NG)| half-tab
                       |   Insert spaces until the next half tab.
                       |
   C-@                 | set-mark-command
   or C-Space          |    Sets the mark at the cursor position, thus
                       |   defining the start (end) of the region.
                       |
   C-g                 | keyboard-quit
                       |   If you have started a complex command
                       |   (described below), but you do not want to
                       |   complete it, this command can be used to
                       |   abort it.
                       |
   C-h                 | [Help prefix]
   C-h a, C-h b,       | apropos, describe-bindings,
   C-h c, C-h d        | describe-key-briefly, list-key-names
   C-h f               | describe-function
                       |   Various help commands
                       |
   C-k                 | kill-line
                       |   Cuts to end-of-line if there is anything
                       |   in it; if not, delete the line itself.
                       |   Two C-k's deletes a line with text.
                       |
   C-l                 | recenter
                       |   Redraws the screen and places the line with
                       |   the cursor in the middle. can also be used
                       |   to clean the screen if it should `mess up'
                       |   for some reason.
                       |
   C-q                 | quoted-insert
                       |   The next character you type in is inserted
                       |   no matter if it is a control character or
                       |   not. If, however, you type C-q and 3 octal
                       |   digits, the appropriate ascii-code will be
                       |   inserted. (e.g. C-q 145' would be equivalent
                       |   to 'e').
                       |
                       |
   C-r, C-s           | isearch-backward, isearch-forward
                       |   Search as you type; and move the cursor to
                       |   the closest occurrence of the string typed
                       |   so far.
                       |
   C-u                | universal-argument
                       |   Type in a number and a command, and that
                       |   command will be executed the appropriate
                       |   number of times. 'C-u 79 -' will for
                       |   instance insert 79 hyphens.
                       |
   C-w, M-w           | kill-region, copy-region-as-kill
                       |   Cut or copy the current region between
                       |   mark and cursor to the kill buffer.
                       |   See also: C-@ : set-mark-command
                       |
   C-y                | yank
                       |   `Paste' the text in the kill buffer. Used
                       |   after cutting with C-k or C-w, or copying
                       |   with M-w.
                       |
   C-x, C-x 4         | c-x prefix, c-x 4 prefix
                       |   Special prefix characters, applied to
                       |   this and other window, respectively.
                       |
   C-x C-c            | save-buffers-kill-emacs
                       |   Exit. If there are modified buffers,
                       |   emacs will ask if you want to save them.
                       |
   C-x C-f            | find-file
                       |   Load a file into a new buffer. This is
                       |   one of the most frequently used commands.
                       |
   C-x C-l, C-x C-u   | downcase-region, upcase-region
                       |   Change the case of the text inside region.
                       |
   C-x C-s            | save-buffer
                       |   Save current buffer!
                       |
   C-x C-w            | write-file
                       |   Save current buffer, and prompt for
                       |   filename. default is `current' name, just
                       |   like C-x C-s
                       |
                       |
   C-x C-x           | exchange-point-and-mark
                       |   Put the cursor at the mark, and set the
                       |   mark where the cursor was.
                       |
   C-x b             | switch-to-buffer
                       |   If editing more than one file, switch to
                       |   another file.
                       |
   C-x i             | insert-file
                       |   Inserts a file into the current text.
                       |
   C-x k             | kill-buffer
                       |   'Quit' this file.
                       |
   C-x o             | other-window
                       |   Puts the cursor in another window, if any.
                       |   see split-window-vertically below.
                       |
   C-x s             | save-some-buffers
                       |   Save all modified buffers
                       |
   C-x u             | undo
                       |   Undo the last operation and pop it off
                       |   the undo stack.
                       |
   M-C-Up     (HK, NG)| <no name>
                       |   Move cursor to previous screen pane
                       |
   M-C-Down    (HK, NG)| <no name>
                       |   Move cursor to next screen pane
                       |
   M-%               | query-replace
                       |   Replace one string with another, and asks
                       |   for confirmation every time.
                       |
   M-^               | delete-indentation
                       |   Join with line above, squish out spaces.
                       |
   M-c, M-l, M-u      | capitalize-word, upcase-word, downcase-word
                       |   Change the case of word under cursor.
                       |
   M-d               | kill-word
                       |   Cut a word to the internal kill buffer.
                       |
                       |
   M-m          (NG)| delete-line
                       |   Kill entire line, regardless of pointer
                       |   position, including trailing newline
                       |   character.
                       |
   M-q              | fill-paragraph
                       |   Reorder the text in this paragraph,
                       |   making the lines fairly adjusted based
                       |   on fill column.
                       |
   M-w              | copy-region-as-kill
                       |   See C-w : kill-region
                       |
   M-x              | execute-extended-command
                       |   Do any emacs command; including the more
                       |   complex ones that are not assigned to any
                       |   key or key sequence. To find the current
                       |   line number, for example, type
                       |   'M-x what-line'.
```

6. LIMITATIONS AND BUGS

The following is a list of some of NotGNU's limitations, and some of its known bugs. If you discover strange behaviors that are not described here, please report these to the author; her address is listed in the introduction.

Size limitations: File size is limited only by the amount of available memory. Although DNOT endeavors to be a small yet fully functional editor, a typical DOS system (640K) will only have 200-250K available for editing (this limit can be extended slightly by disabling the undo stack). On Unix and Windows versions, file size depends on the amount of virtual memory. File names should not contain more than 1000 bytes. Although you can create a buffer with a longer name DOS/Windows versions will truncate the name to use the DOS limit of 8 + 3. Also, DOS/Windows version files are `limited' to 32,767 characters per line.

Limited mouse support in DOS version (dnot.exe):

- o There is no external clipboard available for the DOS version, so mouse operations make use of the internal kill buffer.
- o Double-click operations are not supported.
- o The text in the topmost pane/buffer on the screen cannot be scrolled down by highlighting regions with the mouse, since the mouse cannot move outside the top of the screen.

No TeX, Lisp or Info: These are currently beyond the scope of NotGNU.

No binary file support: NotGNU is designed to run under both Unix and DOS; it reads a file in either format and writes it in desired format. This sometimes means converting ^M^J (carriage-return line-feed) to just ^J (line-feed) or vice-versa.

Timestamp checking and write-file: Timestamp checking will not save you from overwriting a file if you use write-file directly. Perhaps the buffer should have a timestamp when created that can be checked against a file?

Start-up file processing: can be confused. Also, some commands that require a user action will stop it. Note that if you put (set-fg-color) in the start-up file, that Window's dialog boxes will pop up on Windows 3.x/NT, but X11 versions take a command line argument and thus work. In addition, broken syntax may have the incredibly nasty effect of crashing the editor.

Norton Desktop Bugs: Drag and drop to an iconized window, as well as printing with the /p flag, do not work properly with Norton Desktop. Instead, these operations will insert an internal message into the first line in the current buffer.

Some keys are hardwired: The hardwired keys are not perfectly friendly, but it was much easier to do a couple things that way.

Nested load-file does not work: You can not do (load-file foo) inside a start-up file, etc.

TAB with open-rectangle and yank-rectangle: These functions do not correctly account for TAB characters in the text.

Long-line display is simplistic. Highlighting is broken in some cases of extended lines. Pasting is difficult.

Mouse positioning will invalidate line numbers in the mode line. use C-l as a work around.

describe-bindings: 'C-h b', or describe-bindings, does not list meta or ESC key bindings properly.

Iconified NotGNU disables screen saver? Under investigation.

OS/2 anormalities: NotGNU does not process "if OS/2" or "!if OS/2" statements running WNOT.EXE under WinOS2 or DNOT.EXE under OS/2 DOS window mode.

SunOS anormalities: Sun-OS, as well as some other UNIX clones, do not provide the path of the executable in argv[0]; NotGNU will therefore not be able to find the function description file notgnu.fun or any configuration files in this directory. Use a shell-script or alias to invoke NotGNU with full path name to work around.

7. FREQUENTLY ASKED QUESTIONS

Q) When running iconized, 'make' and 'compile' are S-L-O-W!
A) Copy the file COMMAND.PIF to your windows directory, or if you are using an alternative command interpreter, edit its PIF file. See Section 3.5, MESHELL.PIF, COMMAND.PIF and 3.6, Using an Alternative Command Interpreter.

Q) How do I list available commands?
A) Use ``C-h a'' or ``M-x apropos'', then specify your argument. If you want to list all commands, just press RET. For more detailed description of a given command, use ``M-x describe-function'', bound to ``C-h f''.

Q) Why do the menu items look the way they do in the window menu?
A) The commands look like M-x extended commands, and indeed they they are taken out of the same command table the editor uses. No commands exist in the menubar/pulldowns that don't also exist in the editor via keyboard. And no, there will probably not be a mode to list the English equivalents.

Q) How come there are no scroll bars?
A) [Julie:] I'll make them optional when I get them working correctly.

Q) Will there be an extension language?
A) Someday maybe. Never say never, but remember this is a small(ish) editor and not everyone has tons of disk or ram. Remember this answer in regards to features which are outside the mainstream. :)

Q) Is there a way to save macros to disk and/or incorporate them in the NOTGNU.ST file?
A) You can not save a macro per-se, but some commands can be stuffed into a file which looks like the start-up file and be feed to the editor. However, NotGNU's file-parsing is imperfect and so I've not advertised it too much as you could probably easily break it..

Q) What is the file MYMACS.MSG?
A) This is an internal message "batch" file created by NotBoot, to inform WNOT or NTNOT about files to load etc.

Q) [Julie:] On a SUN 4, running OS 4.1.3 and X11R5 the editor runs and appears to be the same as that on the HP. I had thought that the mouse cut and paste worked in these versions as they do for the DOS Windows version, however this is not correct, or I do not understand what to do. I can highlight text by dragging the mouse with the left button depressed, but when the button is released the highlight goes away. Also the middle mouse button always drops what is in the standard X-Windows mouse buffer, not was I thought I was marking in the editor window.

A) Here's what's going on, I think. Most xterms and newer X11 apps don't use cutbuffer-0 anymore. Well, actually xterms seem to look for both, and x cutbuffer 0 has lower precedence. On my HP-UX X11R5, if any xterm or other app has a current selection, the behavior you describe occurs. If I make the xterm lose selection (shows no highlight) then mb1-drag mb2 click pastes as expected. Framemaker however won't look at cutbuffer-0 content, phooie. I'm thinking of `upgrading' notGNU to use the newer X clipboard stuff sometime in the future.

BTW the highlight is `designed' to go away when you up-click from drag. It's due to the difficulty (overhead) of managing the highlight if a change occurs with the region, ie you move dot. You'll note it is already much slower than I'd like to see. You can always temporarily `view' the region by doing M-x show-region.

Q) DOS version doesn't use extended memory. Can it be changed?

A) No and Yes. DOS version now reads bios console size and set-window-size works . If enough people beat me up and I get a 32 bit compiler, other things may also change..

Q) NotGNU sounded interesting, until I read it doesn't have XXXX

A) Thanks for your comment, it will count as a `vote' for further development! :-)

8. UPDATES AND SERVICES

8.1. Updates

The newest release of NotGNU will at any time be available via anonymous FTP to:

ftp.netcom.com, directory /pub/notgnu/*

(Users of WinSocket FTP clients may need to use /pub/tor/notgnu, since the first one is a symblic link). The following files are available:

INFO.TXT	- Information on current status of NotGNU etc.
UPDATES.TXT	- Description of recent changes in NotGNU.
dnotXX.exe	- DOS version, self-extracting
wnotXX.exe	- Windows version, self-extracting
ntnotXX.exe	- Windows NT/Intel version, self-extracting
ntaxpXX.exe	- Windows NT/DEC AXP version, self-extracting
unotXX.tar.gz	- X-windows/Unix curses sources, GNU-zipped tar.
xnotdocsXX.ps.gz	- Reference manual, GNU-zipped postscript.
xnotdocsXX.dvi.gz	- Reference manual, GNU-zipped TeX DVI

The files can also be obtained via e-mail by sending a request to <notgnu-request@netcom.com>, with or more "SEND packet" commands in the body of your message. 'packet' can be one of:

INFO	- General information about NotGNU
UPDATES	- Description of recent changes in NotGNU.
DNOT	- DOS version binaries
WNOT	- Windows version binaries
NTNOT	- Windows NT/Intel version binaries
NTAXP	- Windows NT/DEC AXP version binaries
UNOT	- X-windows/Unix curses sources
DOC-PLAIN	- Reference manual, plaintext
DOC-PS	- Reference manual, postscript
DOC-DVI	- Reference manual, TeX DVI
DOC-TEX	- Reference manual, TeX info

See the next section for more information on notgnu-request.

8.2. NotGNU services

The following three NotGNU mailing lists are available:

* NotGNU-list is a forum for exchange of experiences, problem reports, and general discussions among NotGNU users. Current number of subscribers: ~100.

* The announcement list is used for notification of new releases (major and beta), and other NotGNU related information. Current number of subscribers: ~170.

* Subscribers of the binaries distribution list will receive new DOS and Windows 3.x releases (major and beta) via e-mail as soon as they are out. Current number of subscribers: ~60.

To subscribe to either of these lists, please send e-mail to notgnu-request@netcom.com, with the following syntax in the body of your message:

```
COMMAND [args] [TO|AS your-address@host.domain (Your Name)]
```

`COMMAND' can be one of:

```
SUBSCRIBE list   - Join 'list'
UNSUBSCRIBE list - Leave 'list'
REVIEW list      - List subscribers of 'list' (NotGNU-list only)
WHICH            - Tell what lists you are subscribed to.
CHANGE address   - Update 'address'
SEND packet [n]  - Send 'packet'; [n] is optional splitsize.
INDEX            - List available files
HELP             - Help with syntax
SERVERINFO       - Information on the autoreplyer program
```

`list' can be one of:

```
ANNOUNCEMENT     - Information on updates, bugs, etc.
DISTRIBUTION     - Get new releases via e-mail.
MAILING          - Interactive discussions
ALPHA-LIST       - Notifications of alpha uploads by the author
```

Examples:

```
send dnot 60
subscribe distribution as address@host.domain (Real Name)
change oldaddress to Real Name <newaddress@host.domain>
help
```