CHATEMACS

By: Randy Gobbel (Gobbel.pa)

requires:CHAT, chatemacs.elc (GnuEmacs Lisp program)

This document last edited on August 24, 1987

INTRODUCTION

ChatEmacs, in conjunction with the **chatemacs.elc** module for GnuEmacs, enables use of the mouse for scrolling and selection in GnuEmacs. It also allows use of the META key for escape-prefix commands and automatically switches Chat in and out of Emacs mode when entering and leaving the editor.

DETAILS

After loading ChatEmacs, typing META-char will send an ESCAPE character, followed by the vanilla character. CTRL-META-char sends an ESCAPE followed by CTRL-char. Once ChatEmacs is active, most Emacs commands should require only one keystroke. Since Emacs was originally designed for terminals with a META shift key, this makes the Emacs command set somewhat more regular and easier to remember. For example, scrolling forward and backward will be on CTRL-V and META-V, respectively.

In order to enable mouse actions, first load CHATEMACS.LCOM into Interlisp. After opening a Chat connection and running GnuEmacs, either load chatemacs.elc manually (by giving the ^Xload command), or add the following line to your GnuEmacs init file (.emacs):

```
(load "chatemacs")
```

After loading chatemacs.elc, the title bar on your Chat window should say "Emacs ON". If not, middle-buttoning the "Emacs" menu item in your Chat window will enable mouse events to be sent to Emacs. After ChatEmacs has been activated for the first time, the Chat window's title bar will always indicate whether Emacs mode is on or off. If your mouse clicks don't seem to be taking effect, check the title bar first!

Automatic switching frees the user from having to manually turn ChatEmacs on and off when using Emacs. In most circumstances (see exceptions below) automatic switching will not interfere with other Chat operations, and can be left enabled. Auto-switching is controlled by:

CHATEMACS.SWITCH.ENABLED   [Variable]

When this variable is non-NIL, Chat will respond to a sequence of two consecutive ESCAPEs by toggling the flag that controls mouse event sending. The state of the flag is noted in the window's title bar, just as if the menu command had been executed. CHATEMACS.SWITCH.ENABLED is defaulted to NIL.

auto-switch-enabled   [GnuEmacs Variable]

This variable controls auto-switching on the GnuEmacs side of the Chat connection. If it is non-nil, GnuEmacs will send a switch command when chatemacs.elc is loaded, and another when exited via a ^X-^C command.

Using Emacs with the mouse

The chatemacs.elc module, at the GnuEmacs end of the connection, determines the interpretation of mouse clicks. The current user interface is more complicated than I would like, and suggestions for improvements are welcome.

The most basic operations are fairly simple: left button in a text buffer moves the Emacs "point" to wherever the cursor is pointing. Right button moves the mark (the typein cursor will move for a couple of seconds just to show you where you've just put the mark), and copies the new region to the kill buffer (for use with "shift-select," see below).

Scrolling with the mouse works more or less as in Interlisp, with the scrollbar being the right-hand part of the screen past column 80. Alternatively, holding down the META key makes the entire text area act as a "scrollbar". As in most Envos environments, left button scrolls the line that the mouse is pointing to to the top of the window, right button moves the top line down to the mouse cursor, and middle button "thumbs", taking the vertical displacement of the mouse cursor as an offset into the file (i.e., top line = beginning of file, bottom line = end of file).

Shift- and control- mouse clicks perform editing operations: shift-left copies the contents of the kill buffer to wherever the mouse is pointing (the closest thing to Interlisp shift-select I could come up with). Control-left and control-right kill from point to where the mouse is pointing (sort of like control-select). Control-shift-left moves the mark without copying anything to the kill buffer.

Mouse clicks in the mode line and minibuffer do things that were inherited from il-mouse's ancestor, a package for the BBN Bitgraph terminal. Maybe you will find them useful. They are: The modeline acts like a sideways scrollbar, left=top. In the minibuffer, left button is equivalent to typing META-X, middle button evals an expression you type in, and (beware!) right button suspends Emacs (equivalent to typing ^X^Z).

As mentioned above, the current user interface is sort of, how shall I say, "gnarly." If you have better ideas, please let me (Gobbel.pa) know.