

# Emacs and Free Software

Brian Caffo

# Emacs

---

- To quote the Emacs manual
  - “Emacs is the extensible, customizable, self-documenting real-time display editor.”
  - “ The name “Emacs” was originally chosen as an abbreviation of Editor MACroS.”
- GNU emacs and Xemacs are two (of many) variants of emacs.
- We focus on GNU emacs. GNU stands for **GNU's Not Unix**
- The original emacs was written in 1975 by Richard M Stallman (RMS), the eccentric founder of the Free Software Foundation (FSF) and winner of one of the MacArthur “Genius Awards”
- GNU emacs is one of many programs in the GNU platform initiated by Stallman and others of the FSF

# Free software?

---

While we're on the topic, free software? From [www.gnu.org](http://www.gnu.org)

- “Free software” is a matter of liberty, not price. To understand the concept, you should think of “free” as in “free speech,” not as in “free beer.”
- Free software [...] refers to four kinds of freedom
  - \* The freedom to run the program, for any purpose
  - \* The freedom to study how the program works, and adapt it to your needs. Access to the source code is a precondition for this.
  - \* The freedom to redistribute copies so you can help your neighbor
  - \* The freedom to improve the program, and release your improvements to the public, so that the whole community benefits. Access to the source code is a precondition for this.

# What copyrights constitute free software?

---

Non-free methods of software distribution

1. Proprietary closed source software
2. Freeware - free in the sense of cost
3. Shareware

Free methods of software distribution

1. Public domain
2. Copyleft - requires all modified and extended versions of the program to be free software as well. Examples: GPL, LGPL
3. Free software that does not require derivative works to be free (some X servers are like this)

The GNU people argue that the term “Open Source software” does not-necessarily imply that the software is free.

# Academia

---

Research academics is an especially good place to use and distribute free software.

For the software we use we need

- Extensibility
- The ability to use the software in any way
- Low cost
- The ability to modify and redistribute software

For much of the software we create we need

- Low level of required maintenance/support, preferably handled by the scientific community
- Independent peer review of code
- The ability to publish on the algorithms used in the code

Go to the GNU web page for more information about free software, and software copyrights and patents

# Back to emacs

---

- When you start up emacs it looks for a file called `.emacs` in your home directory for customization
- In windows, the default file is `c:/_emacs`
- `C-g` stands for hold down `control` and press `g`. This command breaks a shortcut that you started accidentally. If you only remember one command to start out with, here it is.
- `M-w` stands for press and release `ESC` then `w`. On most keyboards, it also means press and hold `Alt` then `w`.
- `C-x C-c` exits emacs
- `shift-left mouse` brings up a menu for changing fonts
- Try `shift` and `control` with the other mouse buttons

# Moving around

---

- Arrow keys
- C-a and C-e beginning and end of line (works in bash shell also)
- M-< and M-> beginning and end of file
- PgUp and PgDn work too, equivalently C-v and M-v
- M-x goto-line RETURN n RETURN goes to line n

Note completions work in emacs!

# Windows

---

- $C-x$  2 splits window horizontally
- $C-x$  3 splits window vertically
- $C-x$  o (oh not zero) move between windows
- $C-x$  0 (zero not oh) remove this window from current view
- $C-x$  1 remove all windows except this one

# Files

---

- `C-x f` get file
- `C-x i` insert file
- `C-x s` save file
- `C-x w` save file as
- `C-x d` dir directory
- `C-x C-d` list directory in the annoying way

# Buffers

---

- $C-x$   $b$  select another buffer
- $C-x$   $C-b$  list other buffers
- $C-x$   $k$  kill current buffer
- $C-x$   $K$  nothing

# Getting rid of stuff, copying pasting, formatting

---

- C-k kill and save current line
- C-w kill highlighted region
- C-space start highlighting
- C-x x move to beginning/end of highlighted region
- M-w save highlighted region
- C-y yank back (or paste)
- C-x u or C-x \_ (ctrl-x underscore) undo
- M-q format paragraph

# Searching/replacing

---

- `C-s word search forward for word`
- `C-r word search backward for word`
- `C-x % query replace (REALLY USEFUL)`
- `C-M-s regexp regular expression search forward`
- `C-M-r regexp regular expression search backward`

# Customizing emacs

---

- One way is use emacs' LISP commands in your .emacs file
- Another is to use the top menu (let's do an example)
  1. `options` then  
`customize emacs` then  
`browse customization groups`
  2. **As an example select** `Faces` then  
`Font Lock` then  
`Global Font Lock Mode`
  3. **Click on** `Toggle until it is on` (`non-nil`)
  4. **Click on** `Save for Future Sessions` (writes to your .emacs file)
- Now every time you start emacs, it will automatically color code your syntax (hooray!)

# Major modes

---

- Major modes are part of what makes emacs better than whatever editor that's not emacs that you may be using
- You call a major mode with `M-x mode`
- examples: `M-x latex-mode` and `M-x c-mode`
- REMEMBER TO USE TAB
- When you load a file with a known extension (.tex, .c, .pl) emacs will guess and load the appropriate mode
- You can only be in one Major mode at a time
- ESS (Emacs Speaks Statistics) is a bunch of major modes for S, R, SAS
- Off of the top of my head, there are major modes for:  $\text{\LaTeX}$ , C, R, SAS, python, perl, C++, java, html, lisp, sql, octave, ... (some you have to install yourself)

# Minor modes

---

- `M-x minor-mode` toggles these modes on and off
- `M-x global-font-lock-mode` turns on/off syntax highlighting
- `M-x transient-mark-mode` cause the active region to be highlighted (by `C-space`)
- `M-x auto-save-mode` turns on/off auto save
- `M-x auto-fill-mode` turns on/off line wrap (I prefer line wrap off because I am a `M-q` junkie)
- `M-x line-number-mode` turns on/off line number on the menu bar
- `M-x scroll-bar-mode` turns on/off the scroll bars
- `M-x menu-bar-mode` turns on/off the menu bars

# Rectangles changed my life

---

- C-x r register copy rectangle to register register
- C-x i r register insert rectangle from register register
- C-x r k kill rectangle and store it
- C-x r y insert the last rectangle killed and stored
- C-x r d delete rectangle do not store it
- C-x r o insert a blank rectangle

# Macros are a religious experience

---

- Macros are the key to being lazy
- To define a macro type “C-x (” then a bunch of commands then “C-x )”
- To invoke the last macro defined type C-x e
- To name the last macro defined type  
M-x name-last-kbd-macro
- To execute a named macro type M-x name
- To save the last named macro to a file open that file and type  
M-x insert-kbd-macro RETURN filename RETURN
- (Use your .emacs file if you want the macro to always be defined.)
- Remember to use tab

# A macro example

---

To define an R comment line and save the relevant Lisp commands in our .emacs file

```
C-x (
##-----##
C-x )
M-x name-last-kbd-macro RETURN rline RETURN
M-x rline RETURN
C-x C-f .emacs RETURN
M-x insert-kbd-macro RETURN rline RETURN
```

Now we have the command `rline` always available to us and typing `M-x rline` produces

```
##-----##
```

# Other nifty stuff

---

- `M-x shell` brings up shell mode (a dos or unix prompt)
- In dir mode (`C-x d RETURN`) a `!` executes a shell command on that file. eg: running SAS on a file
- `M-number character repeats character number times.`  
(Try `M-5 c.`)
- `M-x tetris` brings up a game of Tetris
- Try help menu then `emacs psychiatris`
- emacs has an email client (VM), web browser (W3), version control system, news reader, calendar,...
- A complete emacs book is available at [www.gnu.org](http://www.gnu.org)
- Also try “Learning GNU Emacs” from O’reilly