

Emacs

That editor people make jokes about

Timothy

2020-07-05

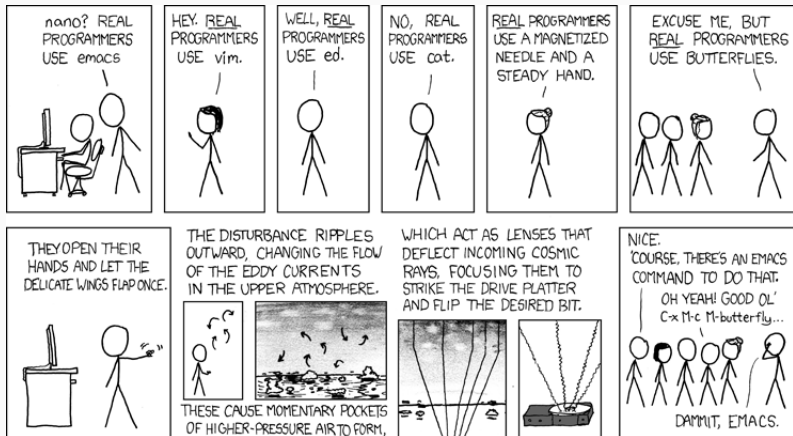
Reputation

Emacs is a great OS, but it lacks a decent text editor.

Alternative acronyms

EMACS is short for “Editing MACroS”, but some fun people have come up with alternatives.

- Eight Megabytes And Constantly Swapping
- Even a Master of Arts Comes Simpler
- Emacs Manuals Are Cryptic and Surreal
- Eradication of Memory Accomplished with Complete Simplicity
- Elsewhere Maybe Alternative Civilisations Survive
- Easily Maintained with the Assistance of Chemical Solutions
- Emacs May Allow Customised Screwups
- Generally Not Used, Except by Middle-Aged Computer Scientists



VIM

usable in just about
any environment.

does one thing, well.



EMACS

flexible, customizable, and
packed with every feature
known to man.

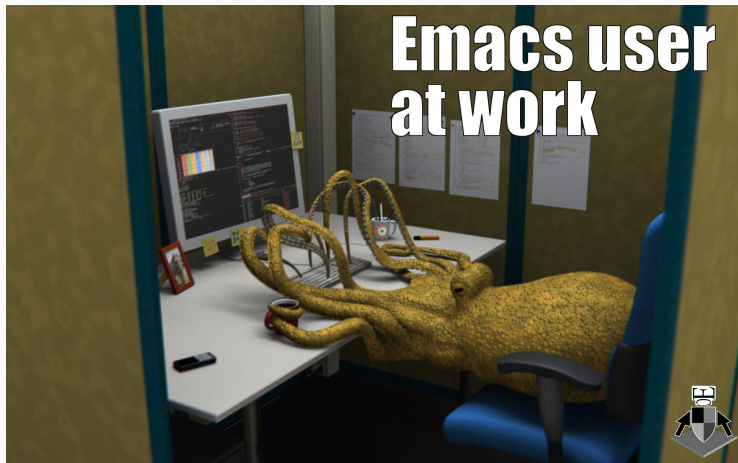


NANO

mostly used by people
who do not know
what they are doing;
or psychopaths.



© 2015 CURTIS LABSAM - CUBE-PRONE.COM



Classical learning curves for some common editors



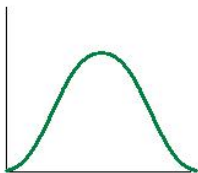
Notepad



Pico



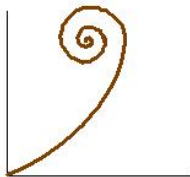
Visual Studio



vi



emacs





So... what is Emacs?

What do the memes tell us?

- Emacs is a thing which edits text
- Emacs gets compared to Vim a lot
- Emacs is versatile, it does a lot
- Emacs is weird

A common misconception

“Emacs is a thing which edits text” . . . so, a text editor?

A common misconception

“Emacs is a thing which edits text” ... so, a text editor? **nope**

A common misconception

“Emacs is a thing which edits text” ... so, a text editor? **nope**



What then?

A definition of Emacs

Emacs is a generic user-centric text manipulation environment.

What then?

A definition of Emacs

Emacs is a generic user-centric text manipulation environment.

A simpler definition

This does sound quite jargony though, let's rephrase.

Emacs is an 'app platform' for anything text

That still sounds like a text editor

Indeed.

That still sounds like a text editor

Indeed. However while that definition matches a text editor, it also applies to a:

That still sounds like a text editor

Indeed. However while that definition matches a text editor, it also applies to a:

- Task planner

That still sounds like a text editor

Indeed. However while that definition matches a text editor, it also applies to a:

- Task planner
- File editor

That still sounds like a text editor

Indeed. However while that definition matches a text editor, it also applies to a:

- Task planner
- File editor
- Terminal Emulator

That still sounds like a text editor

Indeed. However while that definition matches a text editor, it also applies to a:

- Task planner
- File editor
- Terminal Emulator
- Email Client

That still sounds like a text editor

Indeed. However while that definition matches a text editor, it also applies to a:

- Task planner
- File editor
- Terminal Emulator
- Email Client
- Remote server tool (SSH, FTP)

That still sounds like a text editor

Indeed. However while that definition matches a text editor, it also applies to a:

- Task planner
- File editor
- Terminal Emulator
- Email Client
- Remote server tool (SSH, FTP)
- Git frontend

That still sounds like a text editor

Indeed. However while that definition matches a text editor, it also applies to a:

- Task planner
- File editor
- Terminal Emulator
- Email Client
- Remote server tool (SSH, FTP)
- Git frontend
- HTTP client / server

That still sounds like a text editor

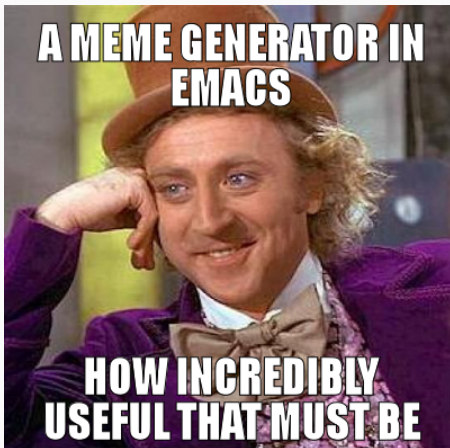
Indeed. However while that definition matches a text editor, it also applies to a:

- Task planner
- File editor
- Terminal Emulator
- Email Client
- Remote server tool (SSH, FTP)
- Git frontend
- HTTP client / server

Emacs can do all of that. Yep.

Oh yes, there's a lot you can do

Ever wanted to [generate memes](#) without leaving your current app?



Emacs is ... perfect?

Nope. Not by a long shot.

Emacs has it's own (large) collection of issues.

- Emacs has annoying quirks
- Areas of Emacs are showing their age (naming conventions, APIs)
- Emacs is single threaded. When something holds that thread up *the whole of Emacs freezes*
- And more... :(

However, Emacs has managed to stay competitive thanks to an unparalleled ability to be tinkered with, and adapt.

Emacs is a Lisp machine

Lisp?

A versatile programming language with just two ingredients

- Atoms
- Lists

Lisp machine?

The core of Emacs is a Lisp dialect, with extensions to support text editing.

How useful is it?

- Emacs has terrible defaults
- Emacs is often a huge time sink for customisation
- Other applications can be seen as an “off the shelf” tool, while Emacs is a raw piece of steel which must be forged and tailored for your needs
 - The effort can be great, but so is the reward

Doom Emacs

What is it?

A fancy configuration base to make Emacs nicer to use and easier to work with.

Duplication of effort

The effort

Just now:

- *Other applications can be seen as an “off the shelf” tool, while Emacs is a raw piece of steel which must be forged and tailored for your needs*
 - *The effort can be great, but so is the reward*

Duplication

Does this mean that for everything you have to start from scratch?

What about common use cases?

Most people want a similar setup, are they doomed to start from nothing, and independently struggle with the same issues?

De-duplication of effort

Doom solves this by providing **modules** for common use cases.
e.g. JavaScript, c++, L^AT_EX, markdown, ...

The first talk

Recall all those nice things I said about modal editing and Vim?

Going to the evil side

Emacs has a fantastic package called EVIL, for “Extensible VI Layer”.

Most popular packages (e.g. `magit`) have an 'evil' companion developed (e.g. `evil-magit`) which adds modal functionality.

From the spacemacs project homepage:

The best editor is neither Emacs nor Vim, it's Emacs and Vim!

Let's get started

Follow the [getting started guide](#). Say yes to every prompt.

https://github.com/hlissner/doom-emacs/blob/develop/docs/getting_started.org

Join the discord server

People there are quite useful. <https://discord.gg/qvGgnVx>

What have you got?

Let's look at how Emacs is now. Open the Emacs app.

Expect it to appear with your other applications. On Linux/Mac you can call emacs in a terminal.

The doom binary

Doom has a executable which is used for some updating doom, installing packages, troubleshooting, and more.

Mac/Linux `~/.emacs.d/bin/doom`

Windows `C:\Users\USERNAME\.emacs.d\bin\doom`

Consider either adding the `bin` folder to your PATH, or (on Mac/Linux) using an alias with:

```
1 alias doom=~/.emacs.d/bin/doom
```


Configuration

Doom has three configuration files.

- `init.el`
- `config.el`
- `packages.el`

On Mac/Linux these will probably be in `~/.doom.d`. On windows try `C:\Users\USERNAME\.doom.d` as long as you set your `HOME`, else try `C:\Users\USERNAME\AppData\Roaming`.

Remember those 'modules' I mentioned?

```
79  :checkers
80  syntax           ; tasing you for every semicolon you forget
81  ;;spell          ; tasing you for misspelling misspelling
82  ;;grammar        ; tasing grammar mistake every you make
```

```
107 :lang
108 ;;agda           ; types of types of types of types...
109 ;;cc             ; C/C++/Obj-C madness
```

```
135 ;;latex         ; writing papers in Emacs has never been so fun
136 ;;lean
137 ;;factor
138 ;;ledger         ; an accounting system in Emacs
139 ;;lua            ; one-based indices? one-based indices
140 markdown        ; writing docs for people to ignore
141 ;;nim            ; python + lisp at the speed of c
142 ;;nix            ; I hereby declare "nix geht mehr!"
143 ;;ocaml          ; an objective camel
144 org              ; organize your plain life in plain text
145 ;;php            ; perl's insecure younger brother
```

1. Uncomment the ones you like the look of — like `\latex`, which we'll be looking at next week.
2. Once you've done that call `doom sync` and all the required packages will be downloaded etc.

Want to read about modules? Press `SPC h d m` and you'll be presented with a list of modules, picking one gives you the documentation.

Basically everything. All your configuration beyond `init.el`. We'll go over some example stuff later.

Any packages you want to use other than those installed from modules.

Read the comments for details, but often you'll use something of the form:

```
1 (package! something)
```

Just remember to run `doom sync`.

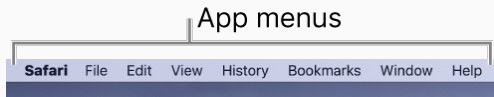
Aside: why are we so obsessed with using a keyboard

This can be explained quite simply. Most people can type 50 wpm. Assuming an average word length of 5, that's 250 keystrokes per minute!

1. Try comparing that to how many actions/clicks you can perform with a mouse.
2. Enabling a fully keyboard-driven workflow means you never have to take your hands off the keyboard.

Leader key

Take a menu bar.



How might you access those items with a keyboard? Perhaps following some trigger you could use **f** to get to *file*, **e** to *edit*, **v** to *view*, etc.

Doom does this, using **SPC** as a trigger.

```
TAB → +workspace  h → +help        q → +quit/session
b → +buffer       i → +insert      r → +remote
c → +code         n → +notes      s → +search
f → +file         o → +open       t → +toggle
g → +git          p → +project    w → +window
```

Basic actions

- `SPC f f` find (and open) file
- `SPC f s` save file
- `SPC b b` switch buffer
- `SPC b k` kill (close) buffer
- `SPC p p` switch project

Fumbling around

The help is great.

- `SPC h` has a lot of help under it
- `SPC h v` look at variables
- `M-x` call an interactive function

Want to set a variable in your config? Add something like this to your `config.el`.

```
1 (setq variable value)
```

Try it out

magit a fantastic git interface

org-mode an exceptionally versatile plaintext format

dired powerful file manager

tramp lets you act on remote servers with ease

It's the language used by everything in Emacs. It's a Lisp, so it probably looks a bit strange. Consider reading a bit about the syntax if/when you feel like configuring Emacs :)