Talking about Emacs for Only 15 Minutes

Aidan Hall

What is Emacs?

- An *Operating System?*
  - You still need a Kernel and init system...
    - for now.
      `https://github.com/a-schaefers/systemE`
      "A lightweight systemd replacement written in Emacs lisp."
- Lacking a decent text editor?
  - EVIL

What *is* Emacs?

The *core* of Emacs performs a relatively small set of highly fundamental operations, notably:

- Buffer management
- Text manipulation
- Window management
- Lisp evaluation
- It's more useful to think about what you can make in Emacs.

Emacs is like a web browser

- What?
- Shoddy, dynamically-typed scripting language
- Focus on document rendering and manipulation
- Applications:
  - Email
  - Note taking
  - Document processing
  - PDF viewers
  - Games
  - Text editor in the browser: `https://vscode.dev`
  - Web browser in the text editor: `eww`

Org Mode

- `* Heading`, `*bold*`, `/italic/`, `_underlined_`, `~code~`, `[[http://destination][link]]`.
- `README.org`
- A fully-featured note taking system
- Project management
- To-do lists and scheduling
- Literate programming (like Jupyter)
- Document formatting
-
```
#+title: Talking about Emacs for Only 15 Minutes
#+author: Aidan Hall
#+options: toc:nil date:nil
#+latex_class_options: [handout]
* COMMENT What is this talk about?
# Getting across the basic concept of what Emacs is.
[[file:discord-response.png]]
# Yeah, light theme /and/ compact mode.
```

Lisp

Lisp syntax is simple: It's just an abstract syntax tree.

-
  ```
  (defvar a (list 1 2 3))
  ```
  ```
  a
  ```

-
  ```
  (car a)
  ```
  ```
  1
  ```

-
  ```
  (cdr a)
  ```
  ```
  (2 3)
  ```

-
  ```
  (cons 0 a)
  ```
  ```
  (0 1 2 3)
  ```

A bigger example

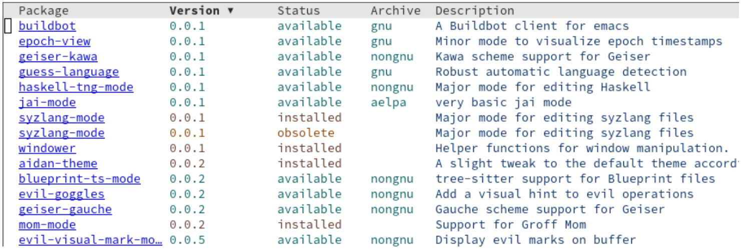
```
(defun fibonacci (n)
  (if (< n 2)
      n
    (+ (fibonacci (- n 1))
       (fibonacci (- n 2)))))

(mapcar #'fibonacci (list 1 2 3 4 5 6 7 8 9 10))
```

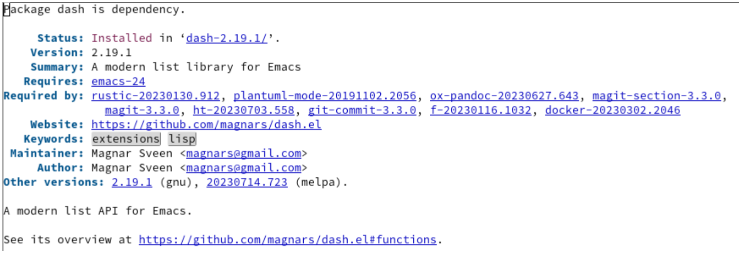
```
(1 1 2 3 5 8 13 21 34 55)
```

Package Manager

"Filetype plugins" and "extensions", but also libraries.



```
  Package              Version ▼     Status       Archive   Description
  buildbot             0.0.1         available    gnu       A Buildbot client for emacs
  epoch-view           0.0.1         available    gnu       Minor mode to visualize epoch timestamps
  geiser-kawa          0.0.1         available    nongnu    Kawa scheme support for Geiser
  guess-language       0.0.1         available    gnu       Robust automatic language detection
  haskell-tng-mode     0.0.1         available    nongnu    Major mode for editing Haskell
  jai-mode             0.0.1         available    aelpa     very basic jai mode
  syzlang-mode         0.0.1         installed              Major mode for editing syzlang files
  syzlang-mode         0.0.1         obsolete               Major mode for editing syzlang files
  windower             0.0.1         installed              Helper functions for window manipulation.
  aidan-theme          0.0.2         installed              A slight tweak to the default theme accord
  blueprint-ts-mode    0.0.2         available    nongnu    tree-sitter support for Blueprint files
  evil-goggles         0.0.2         available    nongnu    Add a visual hint to evil operations
  geiser-gauche        0.0.2         available    nongnu    Gauche scheme support for Geiser
  mom-mode             0.0.2         installed              Support for Groff Mom
  evil-visual-mark-mo… 0.0.5         available    nongnu    Display evil marks on buffer
```



```
Package dash is dependency.

     Status: Installed in 'dash-2.19.1/'.
    Version: 2.19.1
    Summary: A modern list library for Emacs
   Requires: emacs-24
Required by: rustic-20230130.912, plantuml-mode-20191102.2056, ox-pandoc-20230627.643, magit-section-3.3.0,
               magit-3.3.0, ht-20230703.558, git-commit-3.3.0, f-20230116.1032, docker-20230302.2046
    Website: https://github.com/magnars/dash.el
   Keywords: extensions lisp
 Maintainer: Magnar Sveen <magnars@gmail.com>
     Author: Magnar Sveen <magnars@gmail.com>
Other versions: 2.19.1 (gnu), 20230714.723 (melpa).

A modern list API for Emacs.

See its overview at https://github.com/magnars/dash.el#functions.
```

The mode system

- Most user-facing behaviour in Emacs is implemented in modes.
- Modes are ~~just~~ functions

Major modes

Primary functionality, one per buffer.

- `c-mode`
- `shell-mode`
- `mail-mode`
- Inheritance model: derived modes.
  - `fundamental-mode` ← `prog-mode` ← `c-mode`
  - `special-mode` ← `pdf-view-mode`

Minor modes

Secondary functionality, many per buffer or globally.

- `display-line-numbers-mode`
- `evil-mode`
- `auto-fill-mode`

Hooks

A system to automatically run a function when a certain mode activates.

```
(add-hook 'prog-mode-hook 'display-line-numbers-mode)
```

syzlang-mode.el

A major mode for Syzkaller description files.

```
(define-derived-mode syzlang-mode prog-mode
  "Major mode for editing Syzkaller syscall descriptions."
  (setq-local mode-name "Syzlang")
  (setq-local comment-start "#")
  (setq-local font-lock-defaults
              `(((,(rx (or "meta" "define" "include" "resource"
              ↪   "out_overlay" (seq (? "no") "extract")
              ↪   "arches" "incdir" "in" "out" "inout"))
                  . font-lock-keyword-face)))))
```

```
resource fd[int32]: -1
openat$default(fd fd_dir[opt], file ptr[in, filename], flags
↪   flags[open_flags], mode flags[open_mode]) fd
# Almighty!
ioctl$foo(fd fd, cmd int32, arg buffer[in])
```

Thank You

Any questions?