

Delightful Command-Line Experiences

Patrick Callahan

2017-09-14

Outline

- 1 Background
 - About this talk
 - Why use a command-line or textual user interface?
- 2 Command-line basics
 - What (together) comprise a complete command-line environment?
 - How to learn UNIX(-like) command-lines
- 3 How to have a good time
 - Discover things
 - Forget things
 - Configure
- 4 Additional resources and TUI-related topics
 - Lectures, essays, discussions, and manuals

Motivation

Command-line and textual interfaces are:

- available to us (macOS & GNU+Linux, yay!)
- required of us by some of our tools
- widely varied
- highly customizable
- often terrible
 - but they don't have to be!

So why not have an experience that's actually pleasant?

Purpose

My hope is that this talk will

- encourage you to get comfy with a CLI environment that suits you
- provoke you to teach or correct me
- help me get better at doing this

Advantages

Inherent

- composability
- portability

Incidental

- directness
- simplicity
- elegance

Terminal emulator

This is the thing through which you interact with your shell and other terminal applications. It handles keyboard input and sometimes (non-persistent) multiplexing. Examples include:

for macOS

- Terminal(.app)
- iTerm2

for free Unices

- Konsole, Yakuake
- Gnome Terminal, Guake

Terminal emulator

for Windows

- `cmd.exe` GUI
- MinTTY
- ConEmu

Shell

Suitable as a daily driver

- `zsh` - highly extensible, insanely configurable, and largely compatible with `bash` and related shells
- `fish` - modern shell with a strong emphasis on user-friendliness, out-of-the-box functionality, and simple/clear syntax

Shell

Novel and up-and-coming

- `elvish` - very lively, ambitious shell under very active development with a lovely maintainer, novel design, and a wealth of built-ins

Shell

Love yourself— don't use:

- `bash` - de facto standard, present on most Unix(-like) machines, slow, bloated, awful syntax

Terminal multiplexer

- `tmux` - featureful, powerful, popular
- `abduco` & `dvtm` - minimalist, simple, 'suckless'

Command-line utilities

coreutils (the essentials)

- ls, cd, readlink, etc.
- comes in two varieties: GNU and bad

find, sed, awk, grep

- such utility, much textual
- come in two varieties: GNU and bad

Ask the the environment!

- `apropos`: what should I use to do this?
- `whatis`: what command is this?
- `man`: fine, I'll read the manual
- `tldr`: omg just tell me how to `f@#Sing` do the thing!
- `bro`: gimme some common examples (like `tldr`, different contribution model)

Discover from the command-line

Discover files

- `fzf`
- `ripgrep`

Discover online

Curated lists and collections

- My GitHub stars
- Terminals Are Sexy
- Awesome zsh Plugins
- Oh My Fish!
- Fish themes gallery
- Tackle

Discover on an embedded web server

```
omg fish is great you guys
```

- `fish_config`
- `help (fish builtin)`

Forget how you did it (and type less!)

- alias
- abbr
- local scripts

Forget where you were

Via navigation

- `fasd`
- `fzf`
- `fish` loveliness

Via session management

- `tmux-resurrect`
- `iTerm2`

Forget what you did

Via navigation

- `fzf`
- `elvish`

Forget what you needed

By declaring your software selections

- `nix`

By binding your environment to a directory (e.g., for a project)

- `direnv`

By declaring your config

- `dotfiles`

Forget git status!!

At the very least (even if you (God forbid) stick with bash)
get yourself a git prompt.

Configure everything

- crib from public dotfiles
- do everything from scratch

Just kidding! Configure nothing

By relying on a user-friendly shell

- fish
- elvish

A middle path: crowdsource your config

Shell frameworks

- fish: oh-my-fish, tacklebox, chips
- zsh: zplug, antigen, oh-my-zsh (website, GitHub)

Other config management systems

- tmux: tpm - tmux plugin manager

A middle path: crowdsource your config

Public dotfiles

- GitHub does dotfiles
- My dotfiles

Dotfile management

- GNU stow
- super-user-spark
- fresh

More to explore

Lectures

- Spacemagics, or, Editing with Only a Keyboard is Cool
- Literate DevOps: Literate Programming in a World of Command Lines

More to explore

Discourse

- It's time to replace Zsh with a saner shell
- The Elvish philosophy
- A debate about whether to locate certain features in terminal emulators or terminal multiplexers
- Spacemacs documentation on 'discoverability', essential to delightful TUI experiences
- Configurability is the root of all evil (Fish design document)

More to explore

Somehow me

- [Review this presentation outline \(pdf\)](#)