

why

Z shell (zsh)

is *cooler*

than

your shell

A terminal window showing the prompt 'zsh % _' in a monospaced font. The text is white on a black background, with a slight glow effect around the characters.

Brendon Rapp - Cave Lunch #1

why

Z shell (zsh)

is *cooler*

than

your shell

(unless your shell is zsh)

A terminal window showing the prompt 'zsh % _' in a monospaced font on a black background.

Brendon Rapp - Cave Lunch #1

(alternate title)
**she sells
Z shells
by the
sea shore**



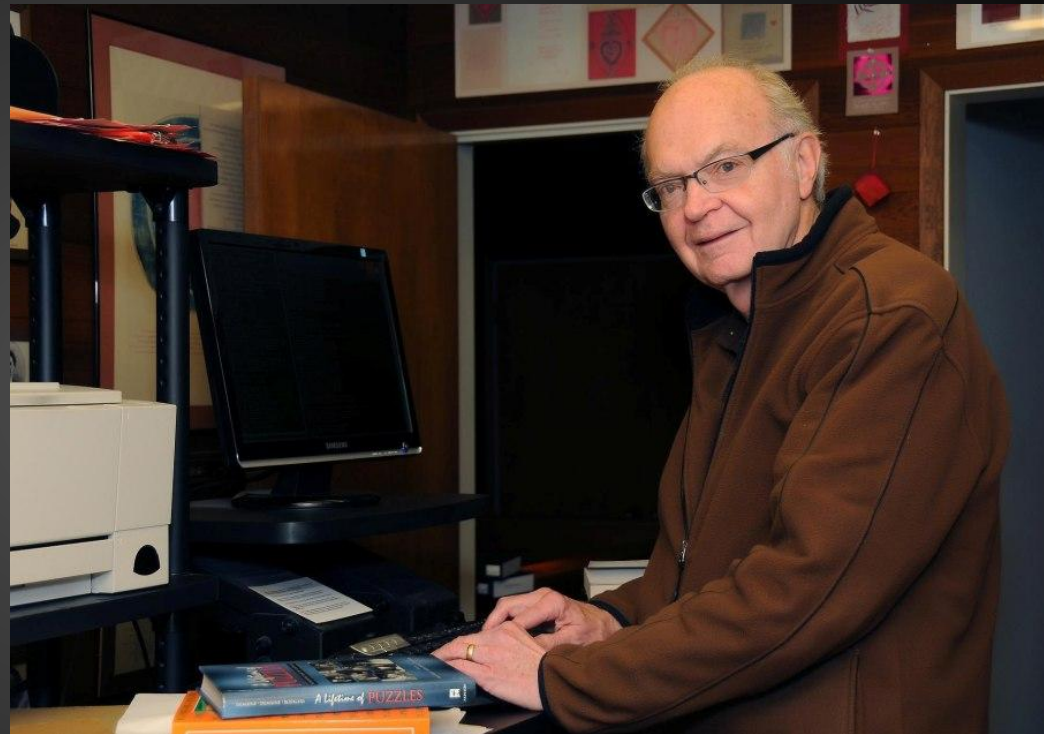
Donald Knuth

Professor Emeritus of
Computer Science at
Stanford

Author of *The Art of
Computer Programming*

"Father of algorithmic
analysis"

Creator of TeX



In 1986, Knuth was asked to write a guest feature for the "Programming Pearls" column in the *Communications of the ACM* journal.

The task was to write a program that would:
read a file of text, determine the n most frequently used words, and print out a sorted list of those words along with their frequencies.

Knuth produced a solution in Pascal that, when printed, was about 10 pages in length. It was well designed, thoroughly commented, and used a novel data structure for managing the word count list.

In response, Doug McIlroy wrote a shell script that produced the same output.

In response, Doug McIlroy wrote a shell script that produced the same output.

McIlroy's script was six lines long.

Doug McIlroy's Shell Script

```
tr -cs A-Za-z '\n' |  
tr A-Z a-z |  
sort |  
uniq -c |  
sort -rn |  
sed ${1}q
```

A brief history of shells

1971: Thompson shell

- Ken Thompson, Bell Labs, first Unix shell
- interactive interpreter, not scripting environment

1977: Bourne shell

- scripting language
- Version 7 Unix, PDP-11
- 1984: *The UNIX Programming Environment*, Kernighan & Pike
- The shell of commercial Unixes
 - System V, AIX, HP-UX, SCO, Solaris, SunOS
 - Still the default on some of these (that are still alive)
- `/bin/sh`
 - compatibility mode in modern shells
 - symlink or hard link to compatible shells in modern Unixes

A brief history of shells

1978: C shell

- BSD Unix
- More "C-like" scripting syntax (kinda)
- Command history
- Aliasing
- tcsh - newer C shell, default on FreeBSD, and OS X systems 10.0-10.2

1983: Korn shell

- Bell Labs (AT&T)
- Proprietary until 2000
- vi and emacs editing modes
- Lots of C shell features
- "middle road" between Bourne and C shell
- pdksh - default on OpenBSD

A brief history of shells

1989: Bourne Again shell (bash)

- GNU, GPL
- first legitimate Free shell (/bin/sh compatible)
 - shells like ksh and csh became Free only much later on
- standard shell for Linux distros, Mac OS X 10.3+
- TAB completion
- extended scripting syntax

1990: Z shell

- most closely resembles Korn shell
- /bin/bash compatibility, drop-in replacement for Bash
- "new" (despite being over 20 years old)
- awesome stuff I'll talk about next

Why use zsh?

**First, a reason that's
kind of lame...**

If you're using Mac OS X...

... your Bash is *old!*

```
bash-3.2$ bash --version
GNU bash, version 3.2.48(1)-release (x86_64-apple-darwin12)
Copyright (C) 2007 Free Software Foundation, Inc.
bash-3.2$
```

(OS X 10.8.2... and many earlier OS X versions too)

OS X: GPL Wasteland

- no GPLv3 on OS X
- OS X bash: final version released as GPLv2
- Homebrew has latest Bash (but many use situation as an excuse to try zsh instead)

Examples of Actual Zsh Awesomeness

zsh: cd completion

```
ubuntu-vm% cd |  
Completing local directory  
Desktop/      Downloads/    Pictures/  
Documents/    Music/       Public/  
Templates/  
Videos/
```

zsh: cd completion

```
ubuntu-vm% cd Desktop/
Completing local directory
Desktop/    Downloads/  Pictures/   Templates/
Documents/  Music/     Public/     Videos/
```

zsh: cd completion

```
ubuntu-vm% cd Documents/
Completing local directory
Desktop/      Downloads/   Pictures/    Templates/
Documents/    Music/      Public/     Videos/
```

bash: cd completion

```
brendon@ubuntu-vm:~$ cd  
.cache/          Downloads/      .mission-control/ Templates/  
.config/         .gconf/        Music/          Videos/  
.dbus/          .gnome2/       Pictures/  
Desktop/        .gvfs/         Public/  
Documents/     .local/        .pulse/  
brendon@ubuntu-vm:~$ cd █
```

bash: cd completion

```
brendon@ubuntu-vm:~$ cd
.cache/           Downloads/       .mission-control/ Templates/
.config/         .gconf/         Music/           Videos/
.dbus/           .gnome2/       Pictures/
Desktop/         .gvfs/         Public/
Documents/       .local/        .pulse/
brendon@ubuntu-vm:~$ cd
.cache/           Downloads/       .mission-control/ Templates/
.config/         .gconf/         Music/           Videos/
.dbus/           .gnome2/       Pictures/
Desktop/         .gvfs/         Public/
Documents/       .local/        .pulse/
brendon@ubuntu-vm:~$ cd █
```

bash: cd completion

```
brendon@ubuntu-vm:~$ cd
.cache/           Downloads/       .mission-control/ Templates/
.config/          .gconf/         Music/           Videos/
.dbus/            .gnome2/       Pictures/
Desktop/          .gvfs/          Public/
Documents/        .local/         .pulse/
brendon@ubuntu-vm:~$ cd
.cache/           Downloads/       .mission-control/ Templates/
.config/          .gconf/         Music/           Videos/
.dbus/            .gnome2/       Pictures/
Desktop/          .gvfs/          Public/
Documents/        .local/         .pulse/
brendon@ubuntu-vm:~$ cd
.cache/           Downloads/       .mission-control/ Templates/
.config/          .gconf/         Music/           Videos/
.dbus/            .gnome2/       Pictures/
Desktop/          .gvfs/          Public/
Documents/        .local/         .pulse/
brendon@ubuntu-vm:~$ cd █
```


zsh: git completion

```
ubuntu-vm% git
Completing alias
'
Completing main porcelain command
add          -- add file contents to index
am           -- apply patches from a mailbox
archive      -- create archive of files from named tree
bisect       -- find, by binary search, change that introduced a bug
branch       -- list, create, or delete branches
bundle       -- move objects and refs by archive
checkout     -- checkout branch or paths to working tree
cherry-pick  -- apply changes introduced by some existing commits
citool       -- graphical alternative to git commit
clean        -- remove untracked files from working tree
clone        -- clone repository into new directory
commit       -- record changes to repository
describe     -- show most recent tag that is reachable from a commit
diff         -- show changes between commits, commit and working tree, etc.
fetch        -- download objects and refs from another repository
format-patch -- prepare patches for e-mail submission
gc           -- cleanup unnecessary files and optimize local repository
```

zsh: git completion

```
ubuntu-vm% git a
Completing main porcelain command
add          -- add file contents to index
am           -- apply patches from a mailbox
archive      -- create archive of files from named tree
Completing interaction command
archimport   -- import an Arch repository into git
Completing plumbing manipulator command
apply        -- apply patch to files and/or to index
```

bash: git completion

```
brendon@ubuntu-vm:~$ git
.bash_history          .gnome2/              .pulse-cookie
.bash_logout          .goutputstream-L361NW Templates/
.bashrc               .gtk-bookmarks       .vboxclient-display.pid
.cache/              .gvfs/               .vboxclient-seamless.pid
.config/             .ICEauthority        Videos/
.dbus/               .local/              .viminfo
Desktop/             .mission-control/   .Xauthority
.dmrc                Music/               .xsession-errors
Documents/           Pictures/            .xsession-errors.old
Downloads/           .profile             .zcompdump
examples.desktop    Public/              .zsh_history
.gconf/              .pulse/              .zshrc
```

bash: git completion

```
brendon@ubuntu-vm:~$ git a  
add          am          annotate      apply        archive  
brendon@ubuntu-vm:~$ git a
```

It's possible to get completion for git (and many other commands) in Bash by installing *bash-completion* package, but the completion is still rudimentary compared to zsh:

- no cycling through options with repeated tabs
- no accompanying info with commands, just a list
- breaks to new prompt line on each tab instead of updating in-place

There may be ways to improve that situation and bring it more in line with zsh, but with zsh, you get it basically out-of-the-box, with a single command in your `.zshrc` to enable completions.

zsh: path expansion

```
ubuntu-vm% cd /u/lo/b
```

(hit TAB, and then...)

zsh: path expansion

```
ubuntu-vm% cd /usr/local/bin/
```

(... the path is expanded in place, provided there is only one path matching that pattern)

zsh: path expansion

```
ubuntu-vm% cd /u/l/b
```

(If there isn't only one distinct match for the pattern...)

zsh: path expansion

```
ubuntu-vm% cd /usr/l/b
```

(... the first TAB will expand up until it hits an ambiguity...)

zsh: path expansion

```
ubuntu-vm% cd /usr/lib  
Completing directory  
lib/      local/
```

(... the next TAB lists the possible matches...)

zsh: path expansion

```
ubuntu-vm% cd /usr/lib/b  
Completing directory  
lib/    local/
```

(... then TAB begins cycling through the possibilities...)

zsh: path expansion

```
ubuntu-vm% cd /usr/local/b  
Completing directory  
lib/    local/
```

(... until you get to the one you want, and hit the Right arrow to "select" it ...)

zsh: path expansion

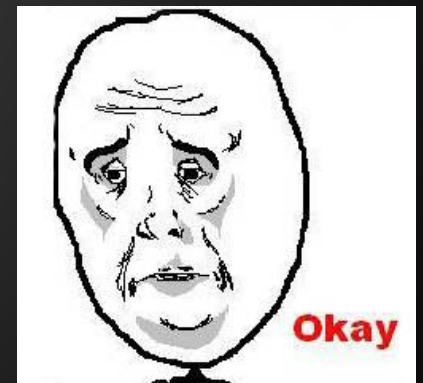
```
ubuntu-vm% cd /usr/local/bin/  
Completing directory  
lib/      local/
```

(... and then TAB resumes matching through the rest of the path)

bash: path expansion?

```
brendon@ubuntu-vm:~$ cd /u/l/b
```

(hit TAB ... nothing)



zsh: path replacement

```
brendon@ubuntu-vm ~ % cd /usr/local/bin  
brendon@ubuntu-vm /usr/local/bin  
% 
```

(Whoops. I meant */usr/local/share*)

zsh: path replacement

```
brendon@ubuntu-vm ~ % cd /usr/local/bin
brendon@ubuntu-vm /usr/local/bin
% cd bin share
/usr/local/share
brendon@ubuntu-vm /usr/local/share
% 
```

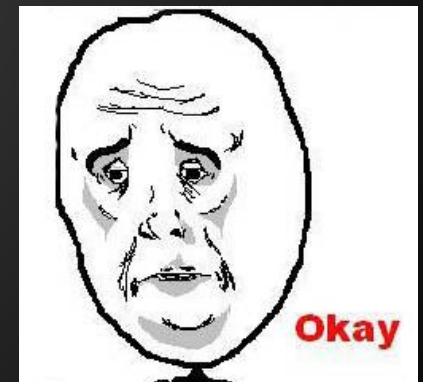
zsh: path replacement

```
brendon@ubuntu-vm ~ % cd /srv/www/site1/current/log
brendon@ubuntu-vm /srv/www/site1/current/log
% cd site1 site2
/srv/www/site2/current/log
brendon@ubuntu-vm /srv/www/site2/current/log
% █
```

(no more "cd ../../../../")

bash: path replacement?

```
brendon@ubuntu-vm:~$ cd /srv/www/site1/current/log  
brendon@ubuntu-vm:/srv/www/site1/current/log$ cd site1 site2  
bash: cd: site1: No such file or directory  
brendon@ubuntu-vm:/srv/www/site1/current/log$ █
```



zsh: right prompt

PROMPT (or PS1): left prompt (like bash)

RPROMPT (or RPS1): right prompt!

zsh: spelling correction

```
brendon@ubuntu-vm ~ % sl  
zsh: correct 'sl' to 'ls' [nyae]? █
```

zsh: spelling correction

```
brendon@ubuntu-vm ~ % sl
zsh: correct 'sl' to 'ls' [nyae]? y
Desktop      Downloads      Music          Public         Videos
Documents    examples.desktop Pictures        Templates
brendon@ubuntu-vm ~ %
```

zsh: spelling correction

```
brendon@ubuntu-vm ~ % vm ./something/with/super/long/path /to/somewhere/else/wit  
h/long/path  
zsh: correct 'vm' to 'mv' [nyae]? █
```

zsh: aliases

Normal aliases:

```
alias ls='ls --color=auto'
```

zsh: aliases

Global aliases - appear anywhere in command string

```
alias -g gp=' | grep -i'
```

```
% ps ax gp ruby
```

```
=> ps ax | grep -i ruby
```


zsh: aliases

Suffix aliases - "Open With..."

```
alias -s rb=vim
```

```
alias -s log="less -MN"
```

```
alias -s html=chromium
```

```
% user.rb
```

```
=> vim user.rb
```

```
% development.log
```

```
=> less -MN development.log
```

```
% index.html
```

```
=> chromium index.html
```

zsh: extended globbing

```
brendon@ubuntu-vm /var/log % ls -l *.log
-rw-r--r-- 1 root  root    0 Jan 25 12:52 alternatives.log
-rw-r----- 1 syslog adm   193 Jan 25 13:17 auth.log
-rw-r--r-- 1 root  root  2158 Jan 25 12:21 boot.log
-rw-r--r-- 1 root  root 47105 Aug 23 11:51 bootstrap.log
-rw-r--r-- 1 root  root    0 Jan 25 12:52 dpkg.log
-rw-r--r-- 1 root  root  2478 Aug 23 11:56 fontconfig.log
-rw-r--r-- 1 root  root    0 Nov 17 00:19 jockey.log
-rw-r----- 1 syslog adm    0 Jan 25 12:52 kern.log
-rw-r----- 1 syslog adm    0 Nov 16 23:21 mail.log
-rw-r--r-- 1 root  root    0 Jan 25 12:52 pm-powersave.log
-rw-r----- 1 syslog adm    0 Nov 16 23:21 ufw.log
-rw-r--r-- 1 root  root 30638 Jan 25 12:32 Xorg.0.log
brendon@ubuntu-vm /var/log % █
```

Normal globbed search

zsh: extended globbing

```
brendon@ubuntu-vm /var/log % ls -l **/*.log
-rw-r--r-- 1 root  root    0 Jan 25 12:52 alternatives.log
-rw-r--r-- 1 root  root    0 Jan 25 12:52 apt/history.log
-rw-r--r-- 1 root  adm     0 Jan 25 12:52 apt/term.log
-rw-r----- 1 syslog adm    193 Jan 25 13:17 auth.log
-rw-r--r-- 1 root  root  2158 Jan 25 12:21 boot.log
-rw-r--r-- 1 root  root 47105 Aug 23 11:51 bootstrap.log
-rw-r--r-- 1 root  root    0 Jan 25 12:52 dpkg.log
-rw-r--r-- 1 root  root  2478 Aug 23 11:56 fontconfig.log
-rw----- 1 root  root  1855 Nov 16 15:04 installer/casper.log
-rw-r--r-- 1 root  root    0 Nov 17 00:19 jockey.log
-rw-r----- 1 syslog adm     0 Jan 25 12:52 kern.log
-rw----- 1 root  root  2072 Jan 25 12:21 lightdm/lightdm.log
-rw----- 1 root  root   986 Jan 25 12:21 lightdm/x-0.log
-rw-r----- 1 syslog adm     0 Nov 16 23:21 mail.log
-rw-r--r-- 1 root  root    0 Jan 25 12:52 pm-powersave.log
-rw-r----- 1 syslog adm     0 Nov 16 23:21 ufw.log
-rw-r--r-- 1 root  root 30638 Jan 25 12:32 Xorg.0.log
brendon@ubuntu-vm /var/log % █
```

Extended globbed search: ****/** = recursive

zsh: environment variable editing

```
brendon@ubuntu-vm ~ % vared PATH  
/usr/lib/lightdm/lightdm:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin  
:/bin:/usr/games
```

zsh: programmable file renaming

```
brendon@ubuntu-vm ~/rename_us % ls
one.txt  three.txt  two.txt
brendon@ubuntu-vm ~/rename_us % zmv '(*).txt' '$1.html'
brendon@ubuntu-vm ~/rename_us % ls
one.html  three.html  two.html
brendon@ubuntu-vm ~/rename_us % zmv '(*).html' 'template_$1.html.haml'
brendon@ubuntu-vm ~/rename_us % ls
template_one.html.haml  template_three.html.haml  template_two.html.haml
brendon@ubuntu-vm ~/rename_us %
```

zsh-history-substring-search

```
brendon@ubuntu-vm ~ % top
```

```
brendon@ubuntu-vm ~ % ps
```

```
brendon@ubuntu-vm ~ % cd ~
```

Down arrow = cycle through command history

zsh-history-substring-search

```
brendon@ubuntu-vm ~ % cd
```

```
brendon@ubuntu-vm ~ % cd ~
```

```
brendon@ubuntu-vm ~ % cd /usr/bin
```

Type part of command and hit up arrow, cycle through only commands that begin with that string

zsh-syntax-highlighting

```
brendon@ubuntu-vm ~ % ne
```

```
brendon@ubuntu-vm ~ % net
```


```
brendon@ubuntu-vm ~ % netsta
```




```
brendon@ubuntu-vm ~ % netstat
```

Highlights valid commands in **green**, invalid commands in **red**

Supports shell commands as well as executables in \$PATH




oh-my-zsh


PUBLIC  **robbyrussell / oh-my-zsh**



 Watch  Star 8,339  Fork 3,202

Code Network Pull Requests 343 Issues 547 Wiki Graphs



A community-driven framework for managing your zsh configuration. Includes 40+ optional plugins (rails, git, OSX, hub, capistrano, brew, ant, macports, etc), over 80 terminal themes to spice up your morning, and an auto-update tool so that makes it easy to keep up with the latest updates from the community. — [Read more](#)
<http://twitter.com/ohmyzsh>



 Clone in Mac  ZIP HTTP SSH **Git Read-Only** `git://github.com/robbyrussell/oh-my-zsh.git`  Read-Only access

 branch: **master** Files Commits Branches 3 Tags

oh-my-zsh /   1000+ commits

Adding MIT-LICENSE Closes #655

 **robbyrussell** authored 10 days ago latest commit 615e41b0ec 

 custom	2 months ago	fixes my wrong commit - fixes #1423 [caarlos0]
 lib	14 days ago	Merge pull request #1077 from Mezzle/add-option-to-show-git-status [robbyrussell]

Other zsh bullet points

- Simple configuration style
- Shared history
 - simple & fast, requires some monkeying to replicate in bash
- Lots of additions for shell scripting
- Output redirection to multiple destinations

And, apparently, plenty of other stuff deeper than I've gotten so far.

The End



Go 49ers