

the z shell (zsh)

Marco Ferraioli

24/10/2015

Linux Day Avellino

marcoferraioli@live.com

marcoferraioli.com

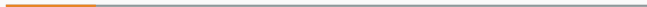
@overflowsystem

The Z Shell (zsh) is cooler than your shell

The Z Shell (zsh) is cooler than your shell
(unless your shell is zsh!)

a brief history of shells

70's



Thompson shell

70's

1971: Thompson shell

1971: Thompson shell

- Was written by
Ken Thompson
(A very Hacker!).

1971: Thompson shell

- Was written by
Ken Thompson
(A very Hacker!).
- Was the first Unix shell.

1971: Thompson shell

- Was written by
Ken Thompson
(A very Hacker!).
- Was the first Unix shell.
- Interactive interpreter.

1971: Thompson shell

- Was written by
Ken Thompson
(A very Hacker!).
- Was the first Unix shell.
- Interactive interpreter.
- No scripting environment.

1971: Thompson shell

- Was written by Ken Thompson (A very Hacker!).
- Was the first Unix shell.
- Interactive interpreter.
- No scripting environment.



1971: Thompson shell

- Was written by Ken Thompson (A very Hacker!).
- Was the first Unix shell.
- Interactive interpreter.
- No scripting environment.



Bourne shell

70's

1977: Bourne shell

70's

1977: Bourne shell

- Replacement for the Thompson shell.

1977: Bourne shell

- Replacement for the Thompson shell.
- Default Unix shell of Unix Version 7.

1977: Bourne shell

- Replacement for the Thompson shell.
- Default Unix shell of Unix Version 7.
- Most Unix-like systems continue to have `/bin/sh`.

1977: Bourne shell

- Replacement for the Thompson shell.
- Default Unix shell of Unix Version 7.
- Most Unix-like systems continue to have `/bin/sh`.
- Scripting language.

1977: Bourne shell

- Replacement for the Thompson shell.
- Default Unix shell of Unix Version 7.
- Most Unix-like systems continue to have `/bin/sh`.
- Scripting language.
- Symlink/Hard link compatible with other modern shells.

C shell

70's

1978: C shell

- BSD Unix.

1978: C shell

- BSD Unix.
- Scripting syntax C-like.

1978: C shell

- BSD Unix.
- Scripting syntax C-like.
- Command history.

1978: C shell

- BSD Unix.
- Scripting syntax C-like.
- Command history.
- Aliasing.

1978: C shell

- BSD Unix.
- Scripting syntax C-like.
- Command history.
- Aliasing.
- tcsh - newer C shell, default on FreeBSD and based distros.

1978: C shell

- BSD Unix.
- Scripting syntax C-like.
- Command history.
- Aliasing.
- tcsh - newer C shell, default on FreeBSD and based distros.
 - On OS-X systems 10.0 - 10.2

80's



Korn shell

80's

1983: Korn shell

80's

1983: Korn shell

- Developed at Bell Labs.

80's

1983: Korn shell

- Developed at Bell Labs.
- Proprietary until 2000.

1983: Korn shell

- Developed at Bell Labs.
- Proprietary until 2000.
- Lots of C shell features.

1983: Korn shell

- Developed at Bell Labs.
- Proprietary until 2000.
- Lots of C shell features.
- Default on OpenBSD.

1989: Bourne Again shell A.K.A. bash

80's

**1989: Bourne Again shell A.K.A.
bash**

80's

**1989: Bourne Again shell A.K.A.
bash**

- GNU, GPL.

1989: Bourne Again shell A.K.A. bash

- GNU, GPL.
- First Free shell (/bin/sh compatible).

1989: Bourne Again shell A.K.A. bash

- GNU, GPL.
- First Free shell (/bin/sh compatible).
- Standard shell for Linux distros, Mac OS X 10.3+

1989: Bourne Again shell A.K.A. bash

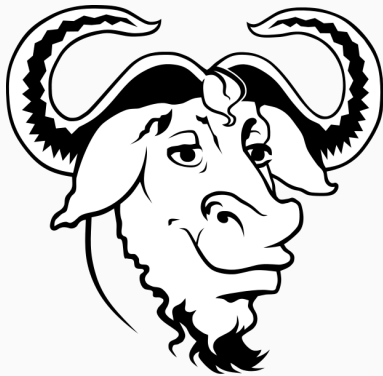
- GNU, GPL.
- First Free shell (/bin/sh compatible).
- Standard shell for Linux distros, Mac OS X 10.3+
- TAB completion.

1989: Bourne Again shell A.K.A. bash

- GNU, GPL.
- First Free shell (/bin/sh compatible).
- Standard shell for Linux distros, Mac OS X 10.3+
- TAB completion.
- Extended scripting syntax.

1989: Bourne Again shell A.K.A. bash

- GNU, GPL.
- First Free shell (/bin/sh compatible).
- Standard shell for Linux distros, Mac OS X 10.3+
- TAB completion.
- Extended scripting syntax.



1989: Bourne Again shell A.K.A. bash

- GNU, GPL.
- First Free shell (/bin/sh compatible).
- Standard shell for Linux distros, Mac OS X 10.3+
- TAB completion.
- Extended scripting syntax.



90's



1990: Z shell A.K.A. zsh

90's

1990: Z shell A.K.A. zsh

1990: Z shell A.K.A. zsh

- /bin/bash compatibility, drop-in replacement for Bash.

1990: Z shell A.K.A. zsh

- /bin/bash compatibility, drop-in replacement for Bash.
- Most closely resembles Korn shell.

1990: Z shell A.K.A. zsh

- /bin/bash compatibility, drop-in replacement for Bash.
- Most closely resembles Korn shell.
- "new" (despite being over 20 years old).

1990: Z shell A.K.A. zsh

- /bin/bash compatibility, drop-in replacement for Bash.
- Most closely resembles Korn shell.
- "new" (despite being over 20 years old).
- The awesome stuff we'll talk about now.

1990: Z shell A.K.A. zsh

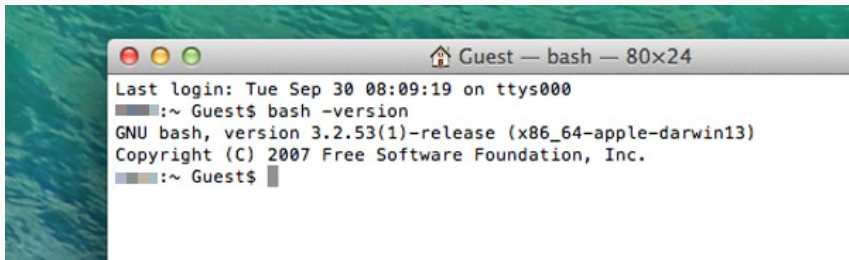
- /bin/bash compatibility, drop-in replacement for Bash.
- Most closely resembles Korn shell.
- "new" (despite being over 20 years old).
- The awesome stuff we'll talk about now.

why use zsh?

First, a reason that's impossible...

If you're using Mac OS X...

...your Bash is *old*!

A screenshot of a macOS-style terminal window. The title bar at the top has three colored window control buttons (red, yellow, green) on the left and a house icon followed by the text "Guest — bash — 80x24" on the right. The terminal content shows a login message: "Last login: Tue Sep 30 08:09:19 on ttys000". Below that, a prompt "Guest\$ " is followed by the command "bash -version". The output of the command is displayed in three lines: "GNU bash, version 3.2.53(1)-release (x86_64-apple-darwin13)", "Copyright (C) 2007 Free Software Foundation, Inc.", and another prompt "Guest\$ " with a cursor. The background of the terminal window is a green and blue marbled pattern.

```
Last login: Tue Sep 30 08:09:19 on ttys000
Guest$ bash -version
GNU bash, version 3.2.53(1)-release (x86_64-apple-darwin13)
Copyright (C) 2007 Free Software Foundation, Inc.
Guest$
```

zsh vs bash

the fight!

zsh vs bash

the fight!

zsh vs bash

- autocomplete

the fight!

zsh vs bash

- autocomplete
 - cd

the fight!

zsh vs bash

- autocomplete
 - cd
 - git

the fight!

zsh vs bash

- autocomplete
 - cd
 - git
 - and other..

the fight!

zsh vs bash

- autocomplete
 - cd
 - git
 - and other..
- path expansion

the fight!

zsh vs bash

- autocomplete
 - cd
 - git
 - and other..
- path expansion
- path replacement

the fight!

zsh vs bash

- autocomplete
 - cd
 - git
 - and other..
- path expansion
- path replacement
- string replacement

the fight!

zsh vs bash

- autocomplete
 - cd
 - git
 - and other..
- path expansion
- path replacement
- string replacement
- spelling correction

the fight!

zsh vs bash

- autocomplete
 - cd
 - git
 - and other..
- path expansion
- path replacement
- string replacement
- spelling correction
- aliases

the fight!

zsh vs bash

- autocomplete
 - cd
 - git
 - and other..
- path expansion
- path replacement
- string replacement
- spelling correction
- aliases
- programmable file renaming

the fight!

zsh vs bash (2)

the fight!

zsh vs bash (2)

- right prompt

the fight!

zsh vs bash (2)

- right prompt
- improved history

the fight!

zsh vs bash (2)

- right prompt
- improved history
- shared history between shells

the fight!

zsh vs bash (2)

- right prompt
- improved history
- shared history between shells
- plugin

the frameworks

the frameworks!

Just a little list

the frameworks!

Just a little list

- alf

the frameworks!

Just a little list

- alf
- dotzsh

the frameworks!

Just a little list

- alf
- dotzsh
- oh my zsh

oh my zsh



oh my zsh

What is?

What is?

- Community-driven framework for managing your zsh configuration

What is?

- Community-driven framework for managing your zsh configuration
- Includes 200+ optional plugins (rails, git, OSX, hub, capistrano, brew, ant, php, python, etc)

What is?

- Community-driven framework for managing your zsh configuration
- Includes 200+ optional plugins (rails, git, OSX, hub, capistrano, brew, ant, php, python, etc)
- Over 140 themes to spice up your morning

What is?

- Community-driven framework for managing your zsh configuration
- Includes 200+ optional plugins (rails, git, OSX, hub, capistrano, brew, ant, php, python, etc)
- Over 140 themes to spice up your morning
- Auto-update tool so that makes it easy to keep up with the latest updates from the community.

What is?

- Community-driven framework for managing your zsh configuration
- Includes 200+ optional plugins (rails, git, OSX, hub, capistrano, brew, ant, php, python, etc)
- Over 140 themes to spice up your morning
- Auto-update tool so that makes it easy to keep up with the latest updates from the community.
- <http://ohmyz.sh/> or <https://github.com/robbyrussell/oh-my-zsh>

install zsh and oh my zsh

install zsh and oh my zsh

How?

install zsh and oh my zsh

How?

- Open Terminal

install zsh and oh my zsh

How?

- Open Terminal
- Install zsh with your package manager (apt-get, pacman, yum, homebrew)

install zsh and oh my zsh

How?

- Open Terminal
- Install zsh with your package manager (apt-get, pacman, yum, homebrew)
- zsh
- `sudo chsh -s /bin/zsh user` (This certainly works on Fedora/Debian/OpenSuse)

install zsh and oh my zsh

How?

- Open Terminal
- Install zsh with your package manager (apt-get, pacman, yum, homebrew)
- zsh
- `sudo chsh -s /bin/zsh user` (This certainly works on Fedora/Debian/OpenSuse)
- For install oh my zsh see the readme.md at:
<https://github.com/robbyrussell/oh-my-zsh> it's very easy

install zsh and oh my zsh

How?

- Open Terminal
- Install zsh with your package manager (apt-get, pacman, yum, homebrew)
- zsh
- `sudo chsh -s /bin/zsh user` (This certainly works on Fedora/Debian/OpenSuse)
- For install oh my zsh see the readme.md at:
<https://github.com/robbyrussell/oh-my-zsh> it's very easy
- Now you will need to edit the `.zshrc` file in your home.

Material:

<https://github.com/paranoiasystem/LinuxDay2015>

The End, Question?