

Worth switching to zsh for casual use? [closed]

Asked 16 years, 3 months ago Modified 10 years, 6 months ago

Viewed 72k times



193



Closed. This question is [off-topic](#). It is not currently accepting answers.

💡 **Want to improve this question?** [Update the question](#) so it's [on-topic](#) for Stack Overflow.

Closed 13 years ago.

[Improve this question](#)

The default shell in Mac OS X is `bash`, which I'm generally happy to be using. I just take it for granted. It would be really nice if it auto-completed *more stuff*, though, and I've heard good things about `zsh` in this regard. But I don't really have the inclination to spend hours fiddling with settings to improve my command line usage by a tiny amount, since my life on the command line isn't that bad.

(As I understand it, `bash` can also be configured to auto-complete more cleverly. It's the configuring I'm not all that keen on.)

Will switching to `zsh`, even in a small number cases, make my life easier? Or is it only a better shell if you put in the time to learn *why* it's better? (Examples would be nice, too `:)`)

@[Rodney Amato](#) & @[Vulcan Eager](#) give two good reasons to respectively stick to `bash` and switch to `zsh`. Looks like I'll have to investigate both! Oh well `:)`

Is there anyone with an opinion from both sides of the argument?

`bash`

`shell`

`zsh`

Share

Improve this question

Follow

edited May 23, 2017 at 11:54



Community Bot

1 • 1

asked Sep 4, 2008 at 8:04



[Will Robertson](#)

64.4k • 33 • 102 • 117

14 Since `zsh` is (almost) a superset of `bash`, you can start with moving to `zsh` and using it like `bash`. Then you can learn the `zsh` gems a little bit at a time. – [orip](#) Dec 21, 2008 at 5:13

If you have a bit of `bash` customization and you often use the shell inside of `emacs`, this does not seem trivial. I posted related question: stackoverflow.com/questions/13326879/... – [justingordon](#) Nov 10, 2012 at 22:58

The *only* thing I dislike about zsh is that (at least on Ubuntu) you'll have to configure it yourself the first time you start it (instead of it providing some sane defaults that you can later on tinker with). If you can get over that, zsh is the best!

– [zrajm](#) Sep 6, 2013 at 20:00

6 Answers

Sorted by:

Highest score (default)



88

Personally, I love zsh.



Generally, you probably won't notice the difference between it and bash, until you want to quickly do things like recursive globbing:



- `**/*.c` for example.



Or use suffix aliases to associate specific progs with different suffixes, so that you can "execute" them directly. The below alias lets you "run" a C source file at the prompt by simply typing `./my_program.c` – which will work exactly as if you typed `vim ./my_program.c`. (Sort of the equivalent to double clicking on the icon of a file.)

- `alias -s c=vim`

Or print the names of files modified today:

- `print *(e:age today now:)`

You can probably do all of these things in bash, but my experience with zsh is that if there's something I want to

do, I can probably find it in [zsh-lovers](#). I also find the book '[From Bash to Z-Shell](#)' really useful.

Playing with the mind bogglingly large number of options is good fun too!

Share Improve this answer

edited Sep 6, 2013 at 20:28

Follow



[zrajm](#)

1,392 ● 1 ● 13 ● 21

answered Sep 17, 2008 at 14:22



[Matt](#)

5,632 ● 5 ● 31 ● 24

1 The first two examples here don't mean anything to me (a casual bash user considering switching to zsh) - could you elaborate on what "recursive globbing" means, or what you mean by "associate specific progs with different suffixes"? – [aaronsnoswell](#) Aug 26, 2013 at 6:28 ✎

5 @aaronsnoswell: "Recursive globbing" means searching a subdirectory *and all of its subdirectories* for files whose names match a particular pattern. Bash doesn't do that by default; you'd `find` or `locate` to search recursively. But once you're letting another program do the searching, you lose the boundaries between names (unless you're careful) as well as access to the shell's internal globbing abilities, and basically have to either make do with what the other program provides, or pipe its output through a program like `grep`, `awk`, or `sed` (all fairly complex in their own right). – [cHao](#) Nov 16, 2013 at 3:14 ✎

6 These days, you can set the `globstar` option in bash to allow for recursive searching as mentioned above. Pretty sure that wasn't the case when this answer was written, though. – [cHao](#) Nov 16, 2013 at 3:19 ✎

With `glob_star_short` in `zsh`, you can now do `**.*`, which is interpreted as `**/*.c`. – PythonNut Mar 23, 2016 at 20:35

I use `zsh 5.3` and get `command not found: age` with `print *(e:age today now:)` – Timo Feb 5, 2018 at 13:34



For casual use you are probably better off sticking with bash and just installing bash completion.

70



Installing it is pretty easy, grab the bash-completion-20060301.tar.gz from <http://www.caliban.org/bash/index.shtml#completion> and extract it with



```
tar -xzvf bash-completion-20060301.tar.gz
```



then copy the `bash_completion/bash_completion` file to `/etc` with

```
sudo cp bash_completion/bash_completion /etc
```

which will prompt you for your password. You probably will want to make a `/etc/bash_completion.d` directory for any additional completion scripts (for instance I have the git completion script in there).

Once this is done the last step is to make sure the `.bash_profile` file in your home directory has

```
if [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
fi
```

in it to load the completion file when you login.

To test it just open a new terminal, and try completing on cvs and it should show you the cvs options in the list of completions.

Share Improve this answer

edited Sep 4, 2008 at 8:36

Follow

answered Sep 4, 2008 at 8:28



Rodney Amato

1,238 ● 9 ● 10

I tried the above, however afterwards whenever I start bash I get 4 "unary operator expected" and 4 "too many arguments" errors. – [AnnanFay](#) Nov 21, 2012 at 18:14

12 If you are using homebrew on OSX you can run `brew install bash-completion`. – [Nathan Villaescusa](#) May 8, 2013 at 18:21

3 Bash-Completion2 is better! On OS X: `brew tap homebrew/versions; brew install bash-completion2` – [Zearin](#) Jan 28, 2014 at 17:21



39



Switch to zsh. You will have access to:

1. `zmv`: You can do: `zmv '(*).mp3' '$1.wma'` for thousands of files.
2. `zcalc`: Extremely comfortable calculator, better than `bc`.
3. `zparseopts`: One-liner for parsing arbitrary complex options given to your script.
4. `autopushd`: You can always do `popd` after `cd` to change back to your previous directory.
5. Floating point support. It is needed from time to time.
6. Hashes support. Sometimes they are just a key feature.

Share Improve this answer

Follow

edited Jan 28, 2014 at 17:23



Zearin

1,481 ● 2 ● 17 ● 37

answered May 26, 2009 at 8:45



anon

19 The example for `zmv` should be `zmv '(*).mp3' '$1.wma'`. Please correct it, such careless presentation can be confusing and frustrating to the inexperienced. – [qed](#) Jan 28, 2013 at 17:48

5 @TravisBear `autoload zcalc; zcalc` (although I myself just type `python` whenever I need a calculator). – [Francisco](#) Jan 29, 2014 at 10:12

10 "autopushd: You can always do popd after cd to change back to your previous directory." Isn't the equivalent of that just cd - ? – [minhaz1](#) Feb 8, 2014 at 3:33

5 `zmv` and `zcalc` are not available out of the box. It's not clear that `zmv` is just renaming files; I thought you might have been converting the format. I don't know what "hashes" support even means. This post seems like it's only "useful" to people that already know zsh. – [mpen](#) Mar 26, 2014 at 17:52

1 @Mark In this case, "hash" is synonymous with "associative array". `bash` 4 was released just a few months prior to this answer. – [chepner](#) Mar 31, 2014 at 14:35



If all you want to use ZSH for is better completion, the configuration is pretty easy. Place this in your `~/.zshrc`:

13



```
autoload -U zutil      # [1]
autoload -U compinit   # [2]
autoload -U complist   # [3]
compinit
```



However, it's worth checking out all the other great features of the ZSH. The above example will give you a pretty plain prompt with good completion. If you don't want to fiddle with configurations, but want to see what ZSH can do for you, Google for "zshrc" and you will get some ready to use configurations to get started.

- [1]: http://zsh.sourceforge.net/Doc/Release/Zsh-Modules.html#The-zsh_002fzutil-Module

- [2]: <http://zsh.sourceforge.net/Doc/Release/Completion-System.html#Initialization>
- [3]: http://zsh.sourceforge.net/Doc/Release/Zsh-Modules.html#The-zsh_002fcomplist-Module

Share Improve this answer

Follow

edited May 28, 2014 at 19:09



Freedom_Ben

11.9k ● 10 ● 77 ● 90

answered Sep 19, 2008 at 23:12



jkramer

15.7k ● 5 ● 50 ● 48

3 What kinds of things will this add completion for? I don't like randomly adding stuff without knowing what it's doing.

– [mpen](#) Mar 26, 2014 at 17:55



6



zsh has a console gui configuration thing. You can set it up pretty quickly and easily without having to fiddle with configuration files. I don't think you will need much time to set it up, probably 10 seconds with just using defaults, so go ahead and try it out.



Share Improve this answer

Follow

answered Sep 4, 2008 at 8:08



Staale

28k ● 23 ● 68 ● 85

10 How do you access this "console gui"? – [mpen](#) Mar 26, 2014 at 17:55

It's usually the first thing you see when you load `zsh` without any related dotfiles. To get it manually, try `autoload -U +X zsh-newuser-install; zsh-newuser-install -f` – [Adam Katz](#) Aug 19, 2015 at 20:23



3



Staale is talking about a wizard like program (CUI) which autoruns the first time you run zsh. Just answer some questions, view/change the defaults and its configured for you.



IBM developerWorks has great resources on zsh.



I have not used very advanced features and so far I have not come across serious differences which should hamper someone coming from bash.

Some examples:

- `!?pattern<Tab>` will autocomplete to the last command in history matching pattern. Very useful.
- You can configure a prompt on the RHS. One use is to keep a fixed width prompt on the left hand side so all commands line up nicely while displaying the `pwd` (or anything of variable width) as the right hand side prompt.
- You can redirect input from multiple files (yet to try this): `cat < file1 < file2 < file3`

Share Improve this answer

answered Sep 4, 2008 at 8:27

Follow



Agnel Kurian

59.4k ● 47 ● 153 ● 225
