

Introduction to git/github

Jonathan Altman

July 13, 2021

<https://github.com/jonathana>

git is not actually a version control system

It is a toolkit for making a distributed version control system (DVCS)

- Originally just a set of primitives (plumbing) created by Linus Torvalds of Linux fame to build a DVCS workflow
- Junio C. Hamano, the original maintainer, had a good feel for how to add complex operations (porcelain) into git to make VCS workflows

What does being a framework mean?

Using git means making choices about how to manage code

- Workflows:
 - git flow
 - feature branch
- Merging:
 - git merge
 - git rebase
- Lots of other choices along the way

**Most of us learn maybe 15% of git's
capabilities and use them 98% of the
time**

git != github

github is git repository hosting + cool workflow features

- github pull requests and issues are magic!
- For (sadly) too many engineers, the difference between git the tool and github hosting and its workflows are not well enough understood
- github is one of many commercial hosting services, there are others -- and you can self host!
- Many people interact with whatever git hosting service via the git command line of their GUI editor's version control tools
- Useful to understand what features are git and which come from your hosting solution

What does being a DVCS mean?

There is no central repository

- git clone to get your local copy does not make "origin" special
- Everything is just a name
- You can decide which repository is the one you synchronize to
- But you really *should* use your git repository hosting service as the central

Being a good teammate

That also means to your future self

- Good commit messages! There is a structure most follow. Do it
- Future you is not as smart as current you
- <https://thoughtbot.com/blog/5-useful-tips-for-a-better-commit-message>

Getting comfortable with git

Tips

- Learn one way of using git as a DVCS. Pick your workflow/merge strategy/etc., or learn the one your team is using
- When you need to get off the path, the internet has the answers
- Understand the magic
 - Hosting service utilities
 - What your tools are doing

Thank you!

Appendix - Links

- git is a scm toolkit: <https://marc.info/?l=linux&m=111293537202443>
- Good commit messages: <https://thoughtbot.com/blog/5-useful-tips-for-a-better-commit-message>
- merge vs. rebase: <https://www.atlassian.com/git/articles/git-team-workflows-merge-or-rebase>
- Example on fixing mistakes: <https://sethrobertson.github.io/GitFixUm/fixup.html>
- Ideas from someone I respect: <https://medium.com/code-adventures/git-conventions-a940ee20862d>
- <https://nfarina.com/post/9868516270/git-is-simpler>
- <http://plasmasturm.org/log/gitidxpraise/>
- <https://tomayko.com/blog/2008/the-thing-about-git>