

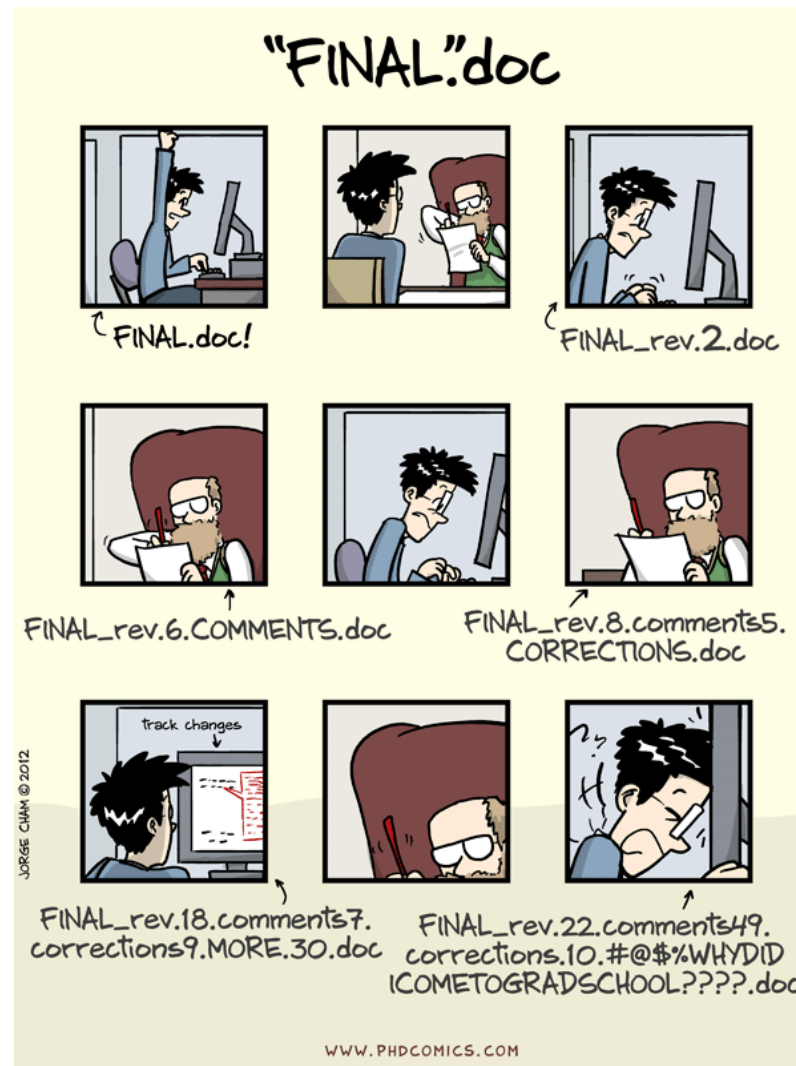


Git and Github introduction

Marco Mambelli – marcom@fnal.gov

14 November 2023

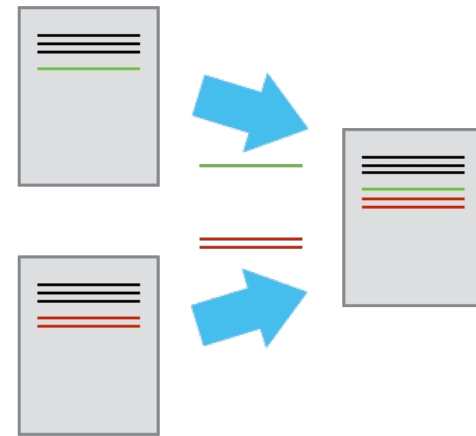
HSF/IRIS-HEP Software Basics Training



“Piled Higher and Deeper” by Jorge Cham, <http://www.phdcomics.com>

Version Control System

- Preserves different version of a document
- Helps merging different contributions



- Answers important questions on the documents
 - What changed?
 - Who changed it?
 - Why?

Image credit: <https://swcarpentry.github.io/git-novice/01-basics/index.html>

Git resources

- Clients for the major platforms
 - **Command line**
 - Fork (free evaluation) <https://git-fork.com/>
 - GitHub desktop <https://desktop.github.com/>
 - GitKraken (free version) <https://www.gitkraken.com/>
 - Tower (30 days trial) <https://www.git-tower.com/>
 - VS code <https://code.visualstudio.com/docs/editor/github>

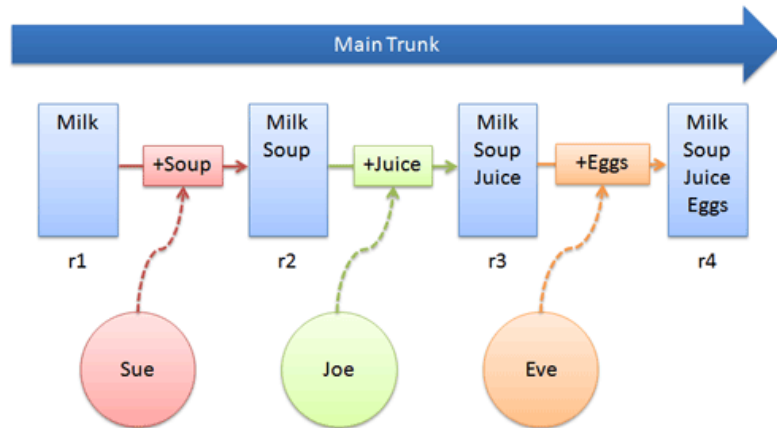
- Online hosting

- Bitbucket
- GitLab
- **GitHub**



Centralized vs distributed VCS

Centralized VCS



- CVS
- SVN

- Mercurial
- **Git**

Distributed VCS

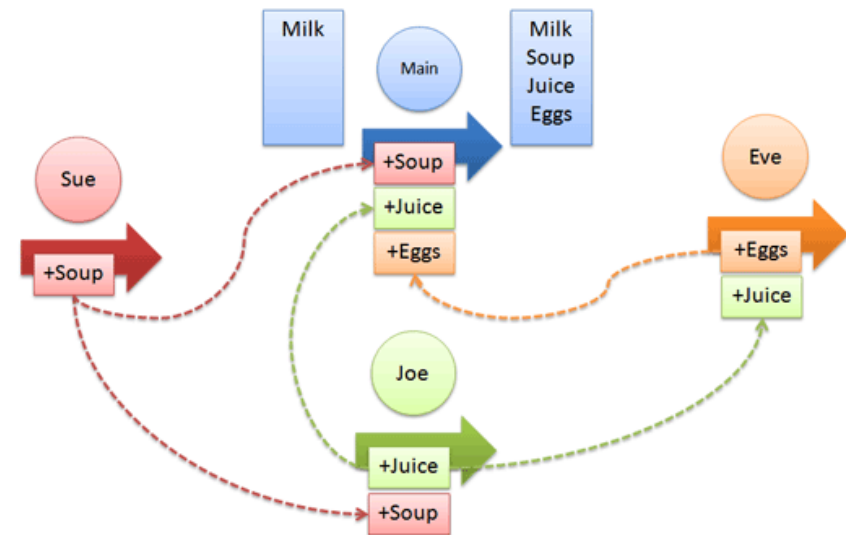
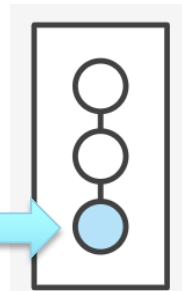
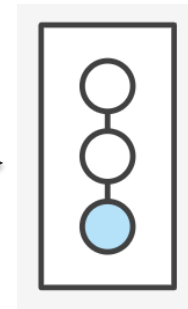
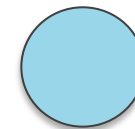
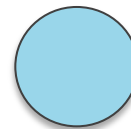


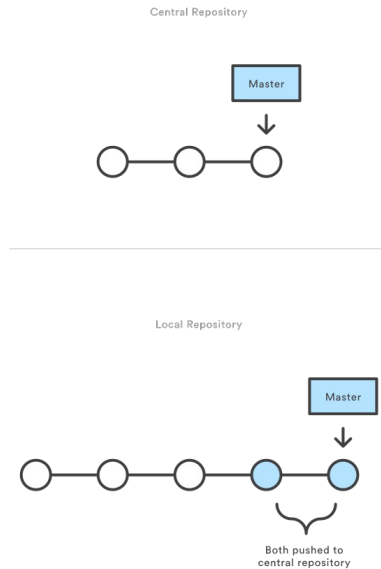
Image credit: <https://betterexplained.com/articles/intro-to-distributed-version-control-illustrated/>

Git concepts – Local repository

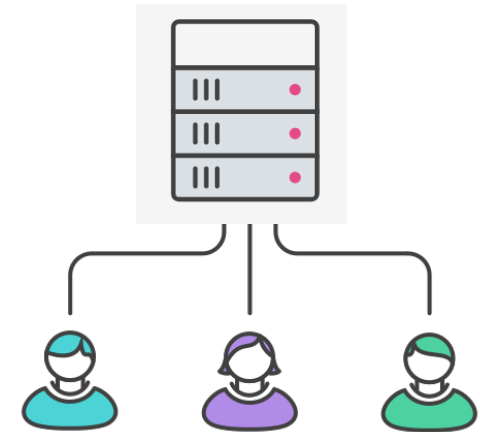
- Snapshot with GUID (SHA1 hash)
 - `git command [sub-command] [options] [arguments]`
- Repository
 - `init`
- Staging
 - `add`
- Commit
 - `commit (checkout)`
- Tag
 - `tag (checkout)`



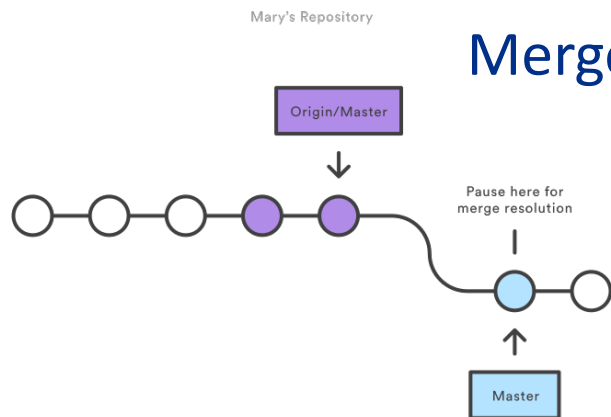
Centralized/Local workflow



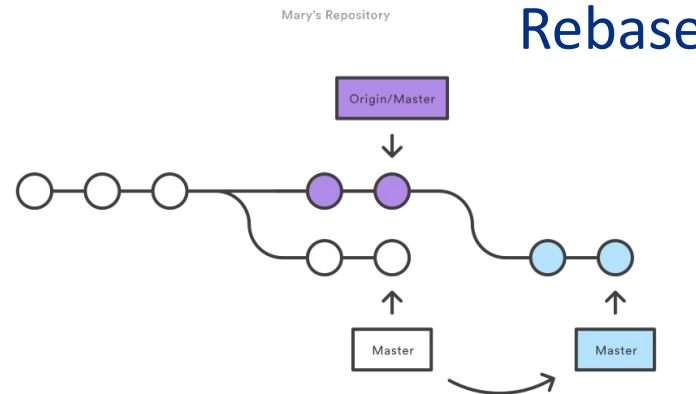
- Single (remote) repo
- Single ordered flow
- Conflicts solved one at the time by the developer



Conflicts resolution



Merge



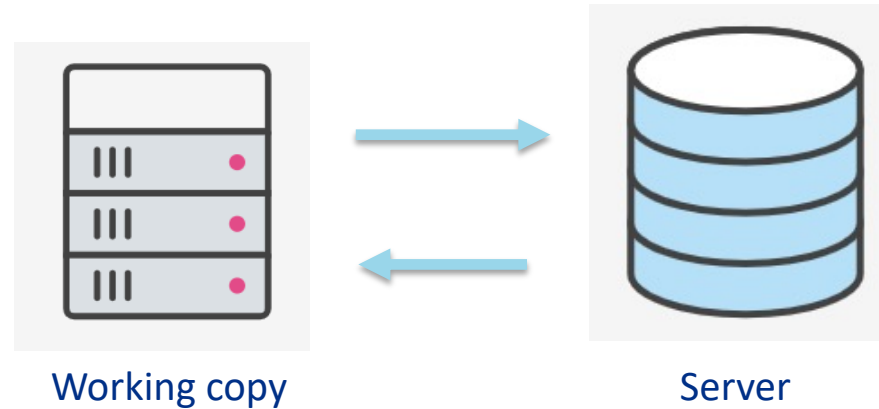
Rebase

Images credit: <https://www.atlassian.com/git/tutorials/comparing-workflows>

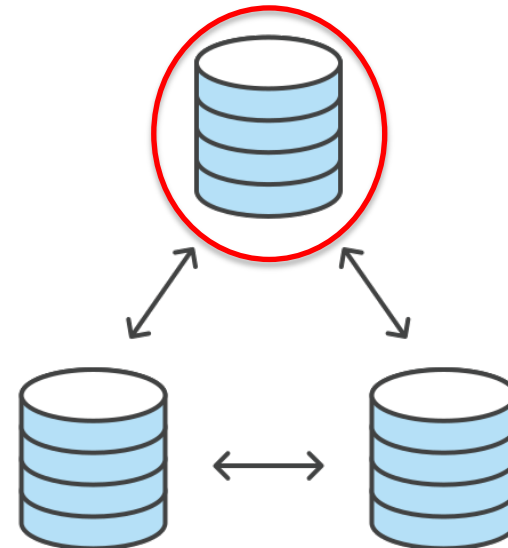
Git remotes – Remote repositories

- Remote

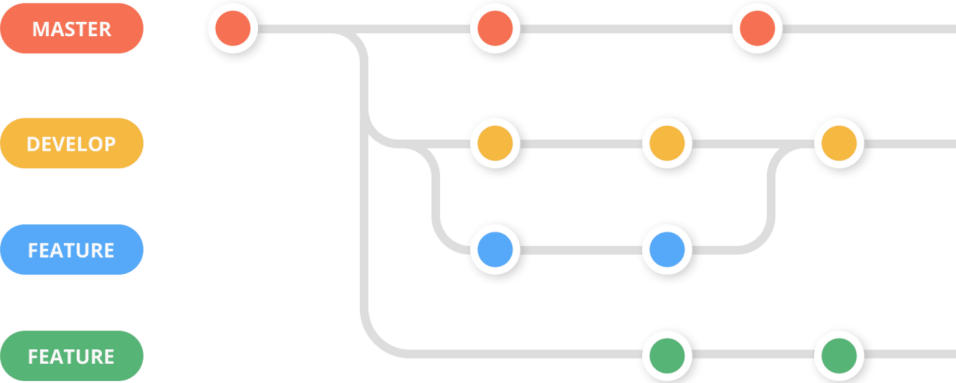
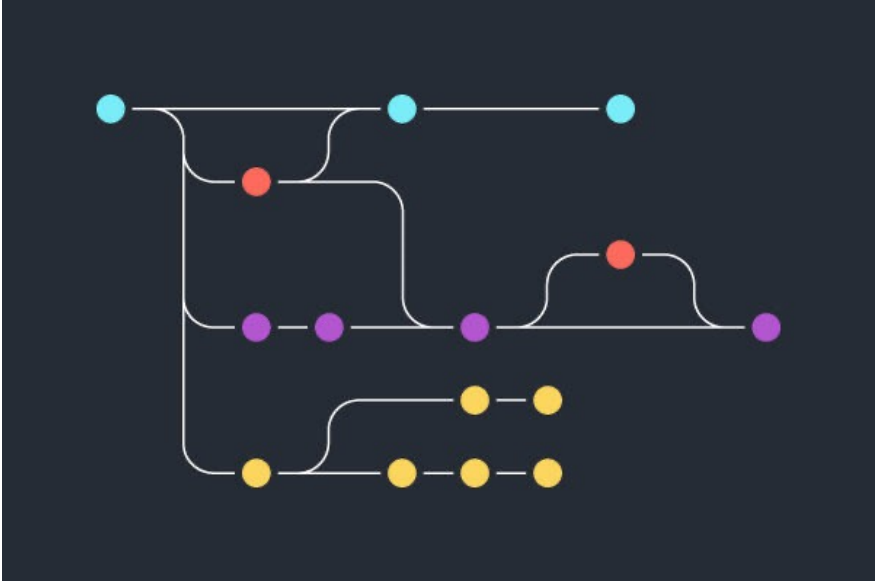
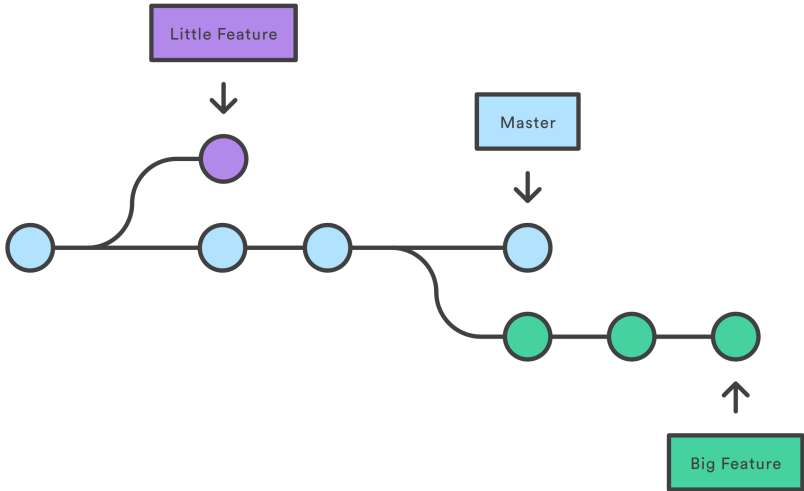
- clone
- remote add
- push
- fetch (pull)
- Local branches track remote ones (`branch -u`)



- Define a main repository!

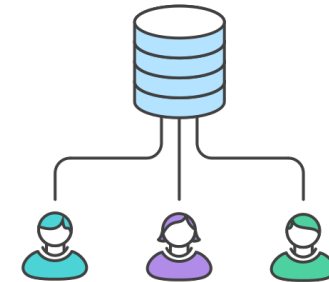
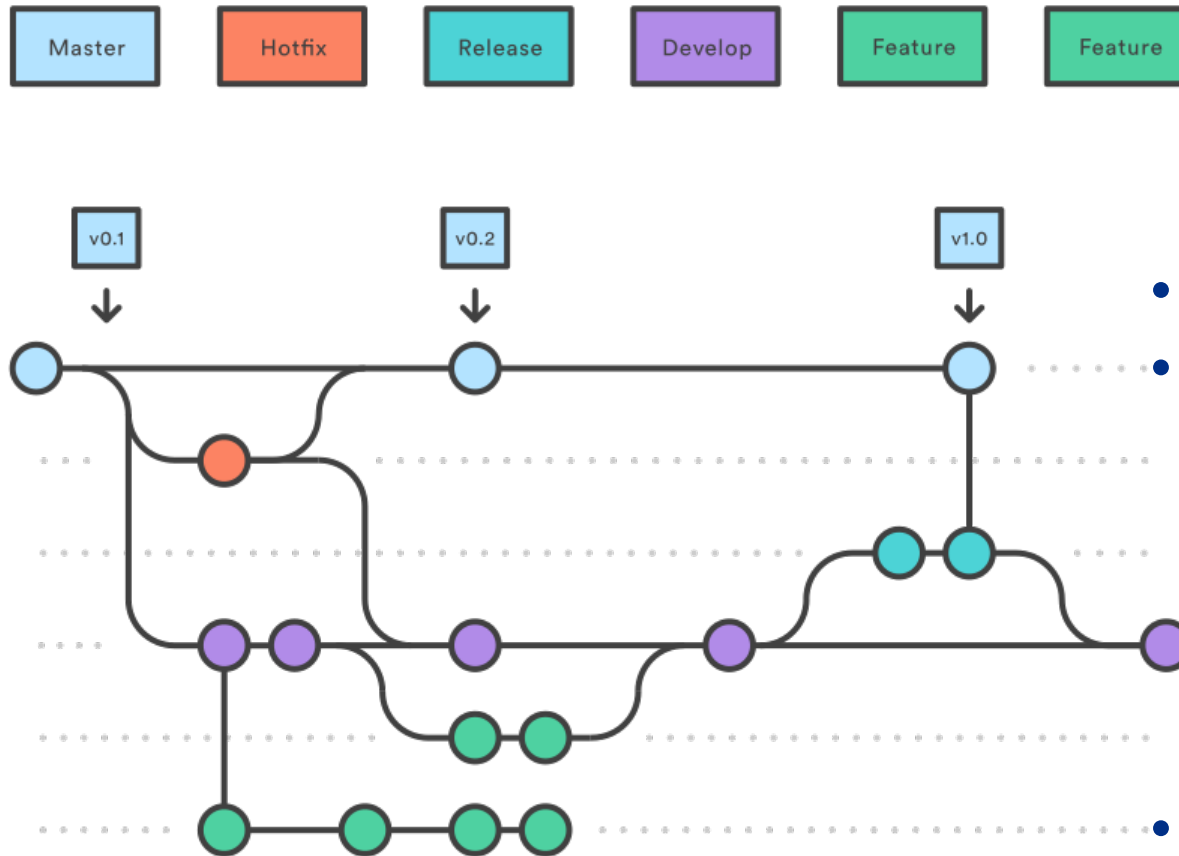


Branches



<https://learngitbranching.js.org/> gamified lessons to learn git branching

Feature branching workflow

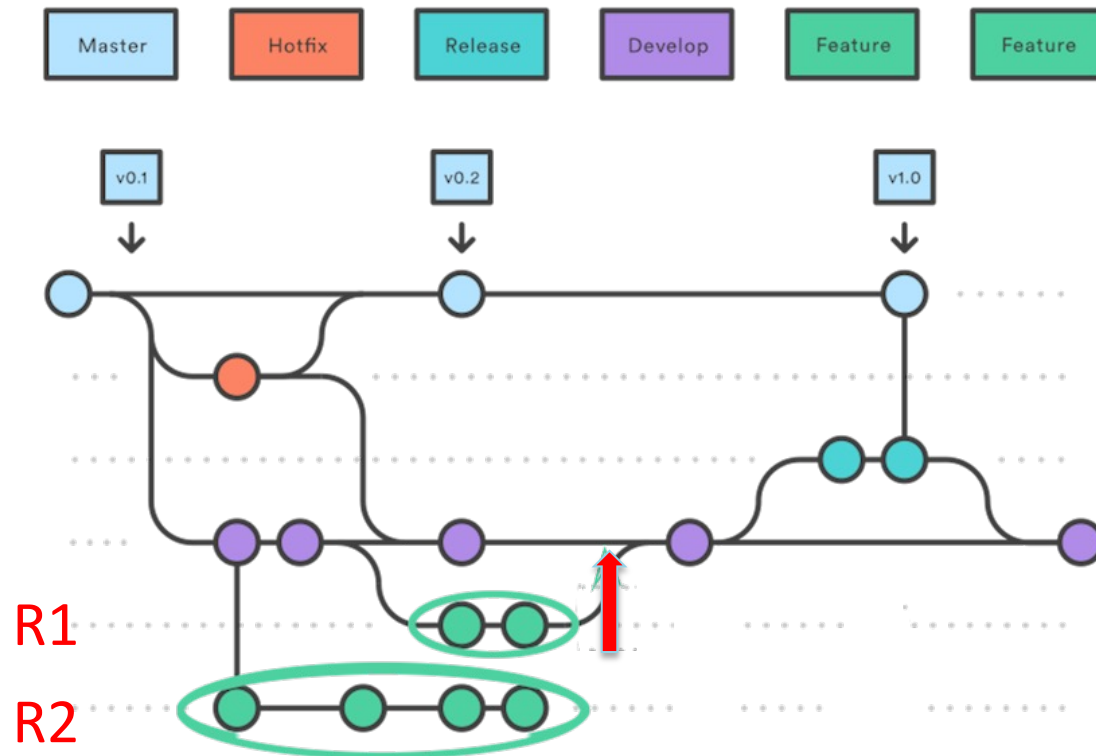
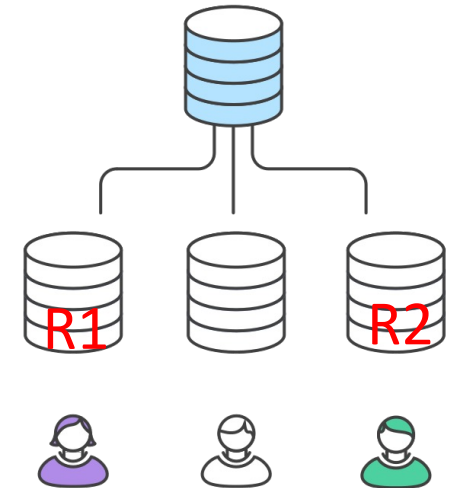


- Single (remote) repo
- Leverage branches:
 - master (releases)
 - Development
 - Features
 - Hotfixes
- Easier to enforce policies

<https://nvie.com/posts/a-successful-git-branching-model/>

Images credit: <https://www.atlassian.com/git/tutorials/comparing-workflows>

Fork and branch workflow



- Multiple forked repos
- Leverage branches
- Feature branches in forked repos
 - Squash
 - Rebase
- Pull requests
- Even easier to enforce policies
- Restricted access

Images credit: <https://www.atlassian.com/git/tutorials/comparing-workflows>

GitHub in brief

- Git repository hosting service <https://github.com>
- Numbers
 - over 40 million users
 - > 100 million repositories (> 28 million public)
 - GitHub, Inc. subsidiary of Microsoft
- Main services
 - Web interface
 - Wiki
 - Github Pages, websites <https://username.github.io>
 - Github Actions <https://pages.github.com/>
 - Pull requests w/ review and comments
<https://help.github.com/en/github/managing-your-work-on-github/about-issues>
 - Integrations

GitHub Desktop and CLI

- GitHub Desktop is a convenient GUI for GitHub
 - <https://docs.github.com/en/desktop>
- GitHub Command Line Interface (CLI) adds GitHub specific functionalities to the command line
 - Easy token authentication w/ “gh login”
 - <https://docs.github.com/en/github-cli>
 - <https://docs.github.com/en/github-cli/github-cli/quickstart>
 - <https://github.com/cli/cli>

Sources and references

- Google Colab and GitHub
 - <https://colab.research.google.com/github/googlecolab/colabtools/blob/master/notebooks/colab-github-demo.ipynb>
 - <https://colab.research.google.com/github/googlecolab/colabtools/blob/master/notebooks/colab-github-demo.ipynb>
 - <https://towardsdatascience.com/google-drive-google-colab-github-dont-just-read-do-it-5554d5824228>
- Git documents and tutorials
 - <https://help.github.com/>
 - <https://www.atlassian.com/git/tutorials/>
 - <https://git-scm.com/>
 - <https://swcarpentry.github.io/git-novice/>
 - <https://education.github.com/git-cheat-sheet-education.pdf>
 - <https://learngitbranching.js.org/>

Important Git recommendations

- One concept, one commit
- Write meaningful commit messages
 - First line is the summary
 - Enough detail to understand the changes
- Is OK to use a GUI
- No PASSWORDS (or keys, PII, ...) in your repo!
- Public software should have a license
 - LICENSE (text file in the root of the repository)
 - BSD 3-clause, Apache 2.0, GitHub has examples
- A DOI, Digital Object Identifier, can facilitate citations
 - <https://about.zenodo.org/>