Git: The Version Control System

Junru Zhong

April 24, 2018

In associate with UIC HCC Computer Association

Contents

- Version Control
- Git
- Remote Git
- Cooperating with Git

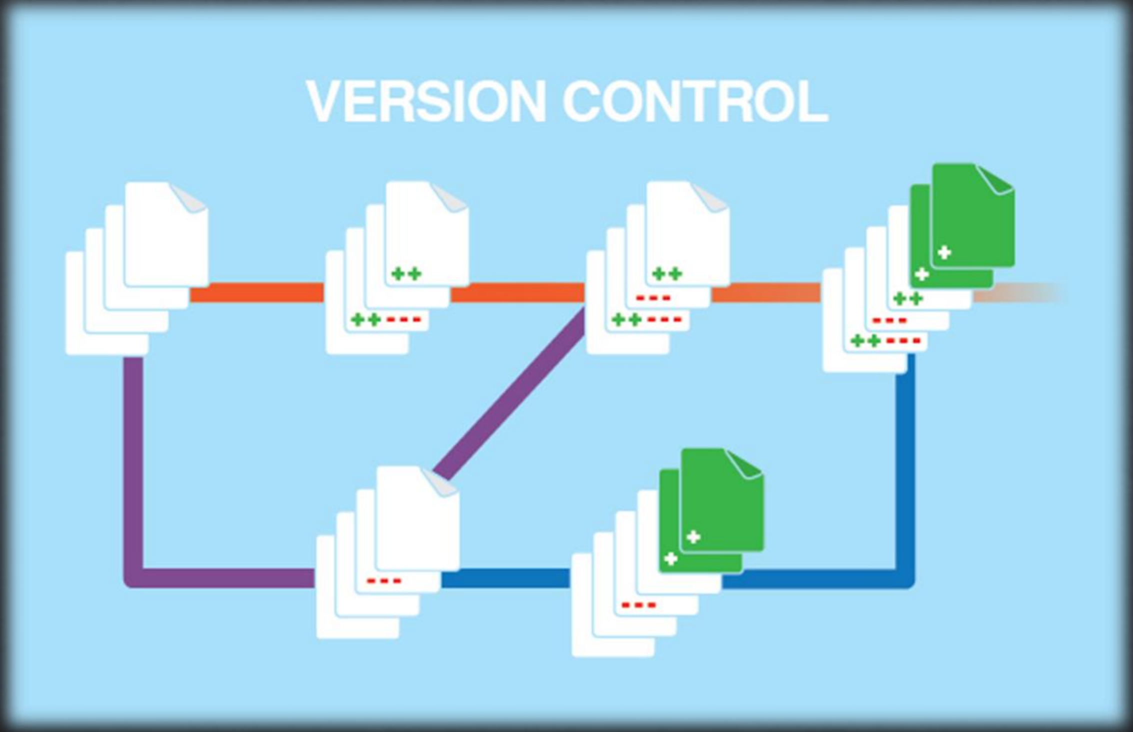


Image source: https://blog.learntoprogram.tv/wp-content/uploads/2016/05/version-control.png

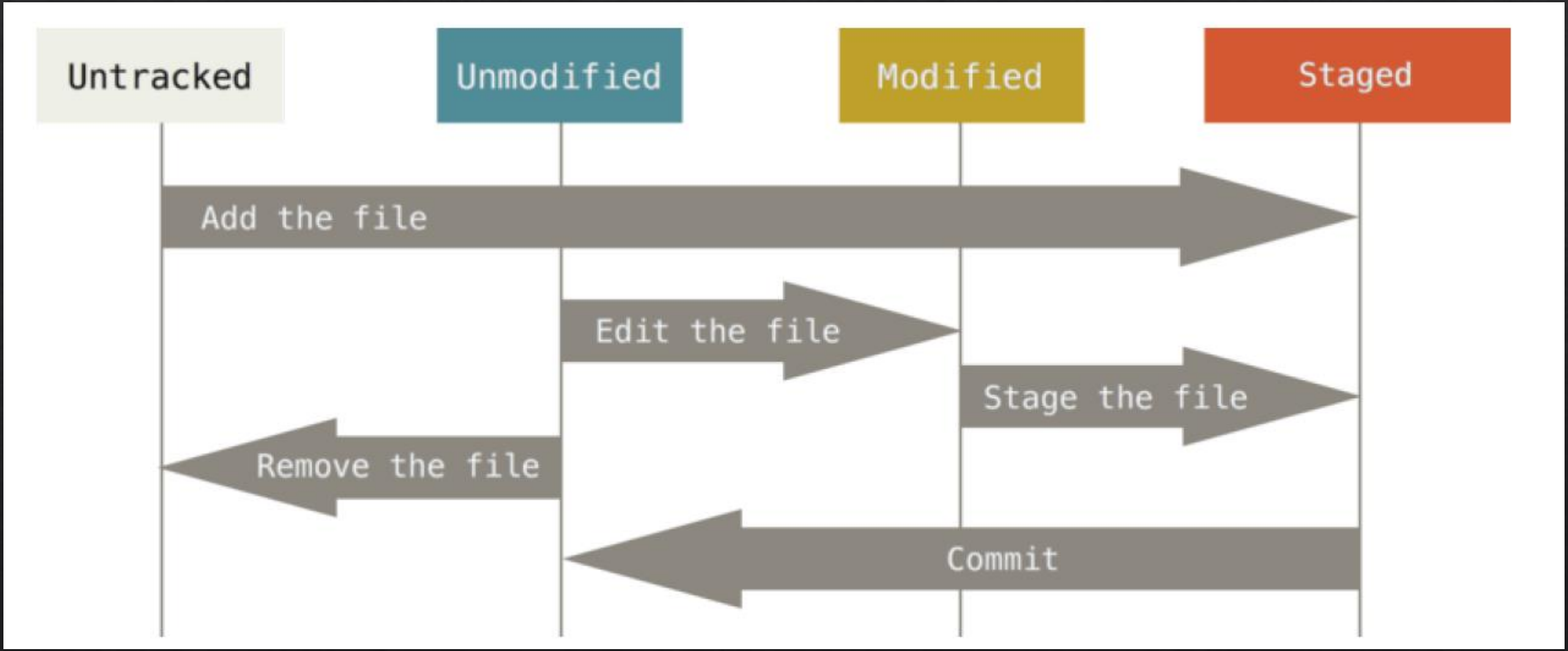
Examples of Version Control

VCS: Manage differences

Git: The Version Control System

Cat proof!

Git basic



Credit: S. Chacon & B. Straub, Pro Git (2nd Edition), 2014

Git basic

- Repository (仓库): where tracked files are stored.
- Commit (提交): tell Git to store this change.
- Staging area (暂存区): changes will be committed.

- Basic operations:
  - `$ git init  #Initialize a repository in current directory.`
  - `$ git add [file]  #Put changed file(s) to staging area.`
  - `$ git commit -m “[message]”  #Commit a change.`

Git may fail...

We need backups!

Data backup 3-2-1 principle: 3 backups, 2 media, 1 remote

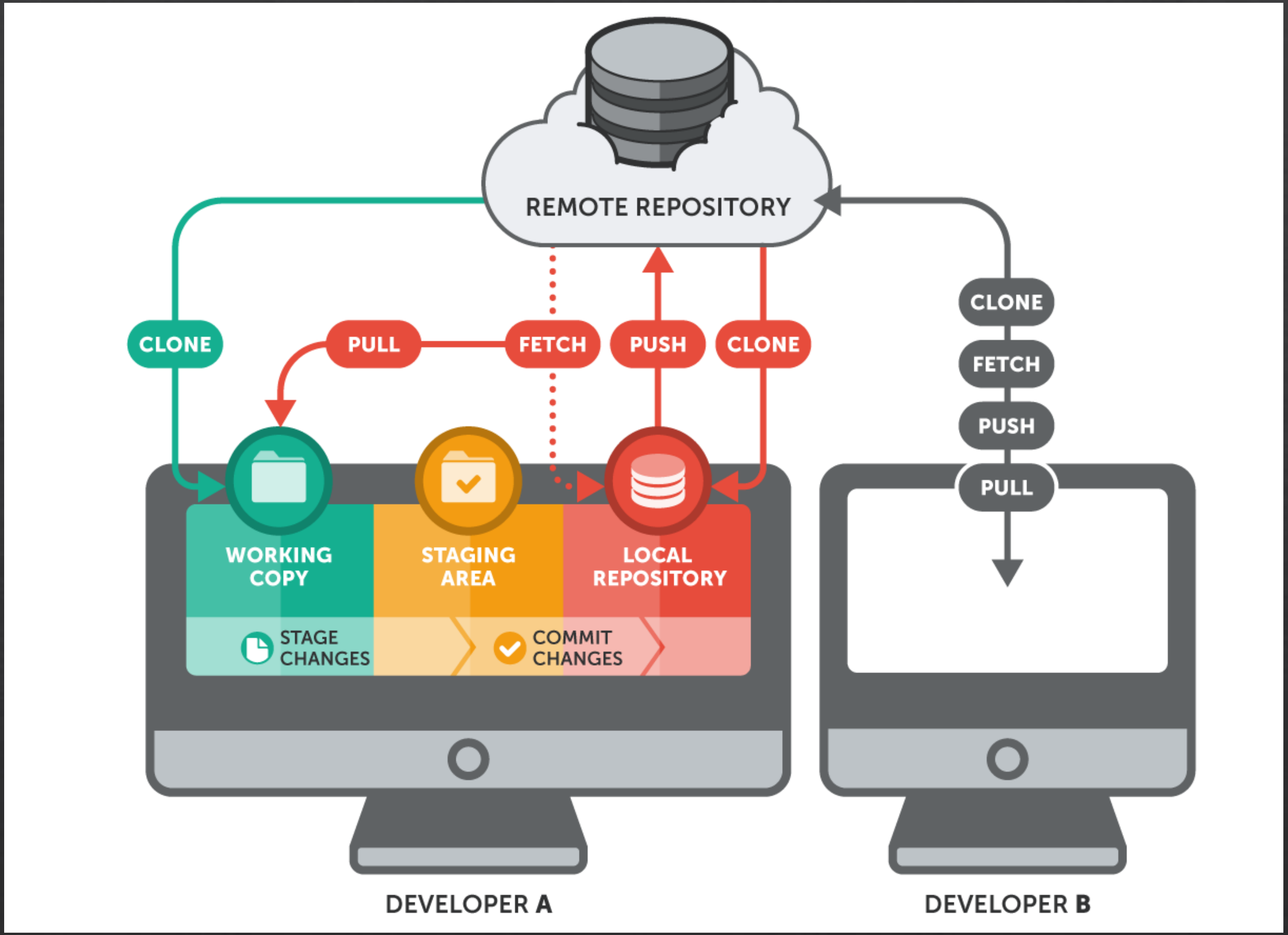
Remote Git Repository

Backup & cooperation

Remote Git basic

- All repositories are same.
- Always pull before push.

- Basic operation
  - `$ git clone [URL]`
  - `$ git fetch`
  - `$ git pull`
  - `$ git push`



Cooperating

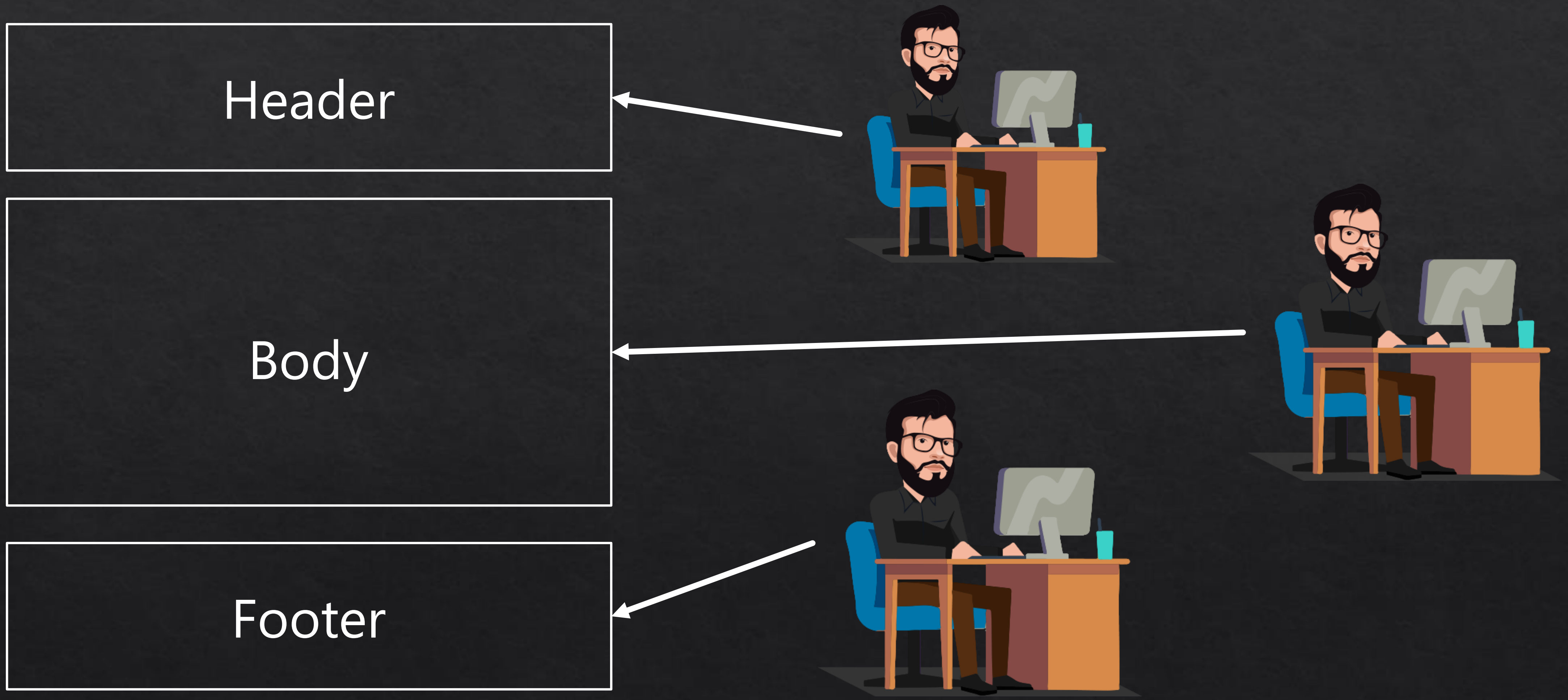


Image source: https://www.vervelogic.com/images/img/hire-developer/ishank-sir.png

Cooperating

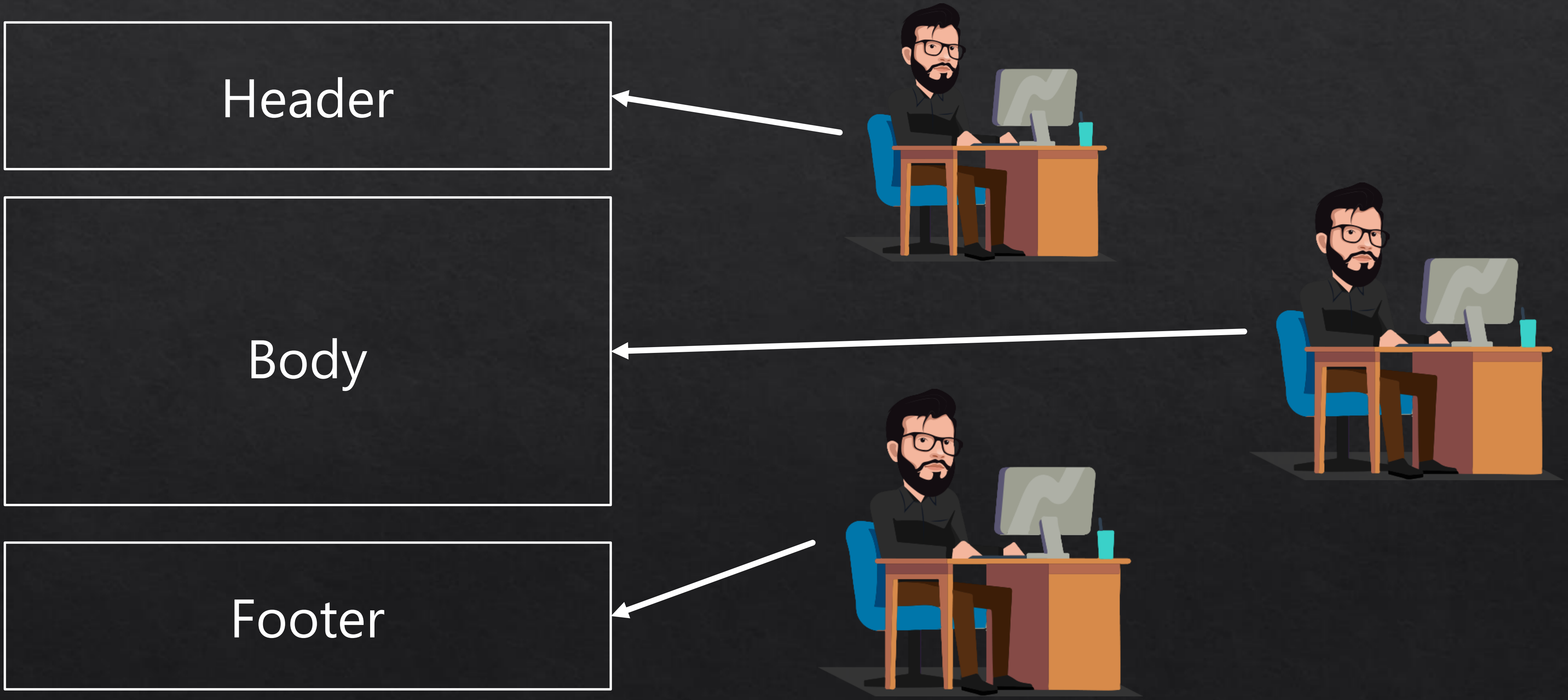


Image source: https://www.vervelogic.com/images/img/hire-developer/ishank-sir.png

Cooperating

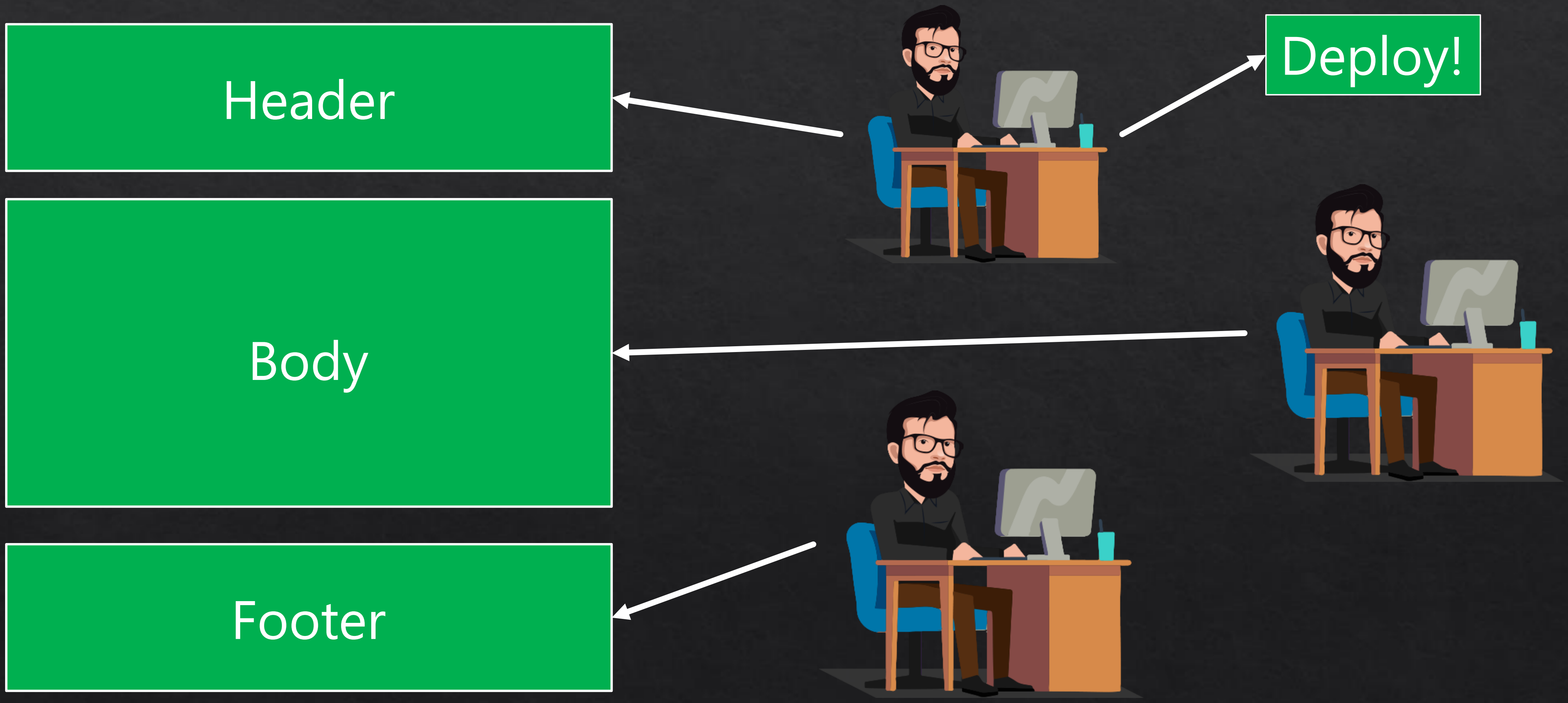
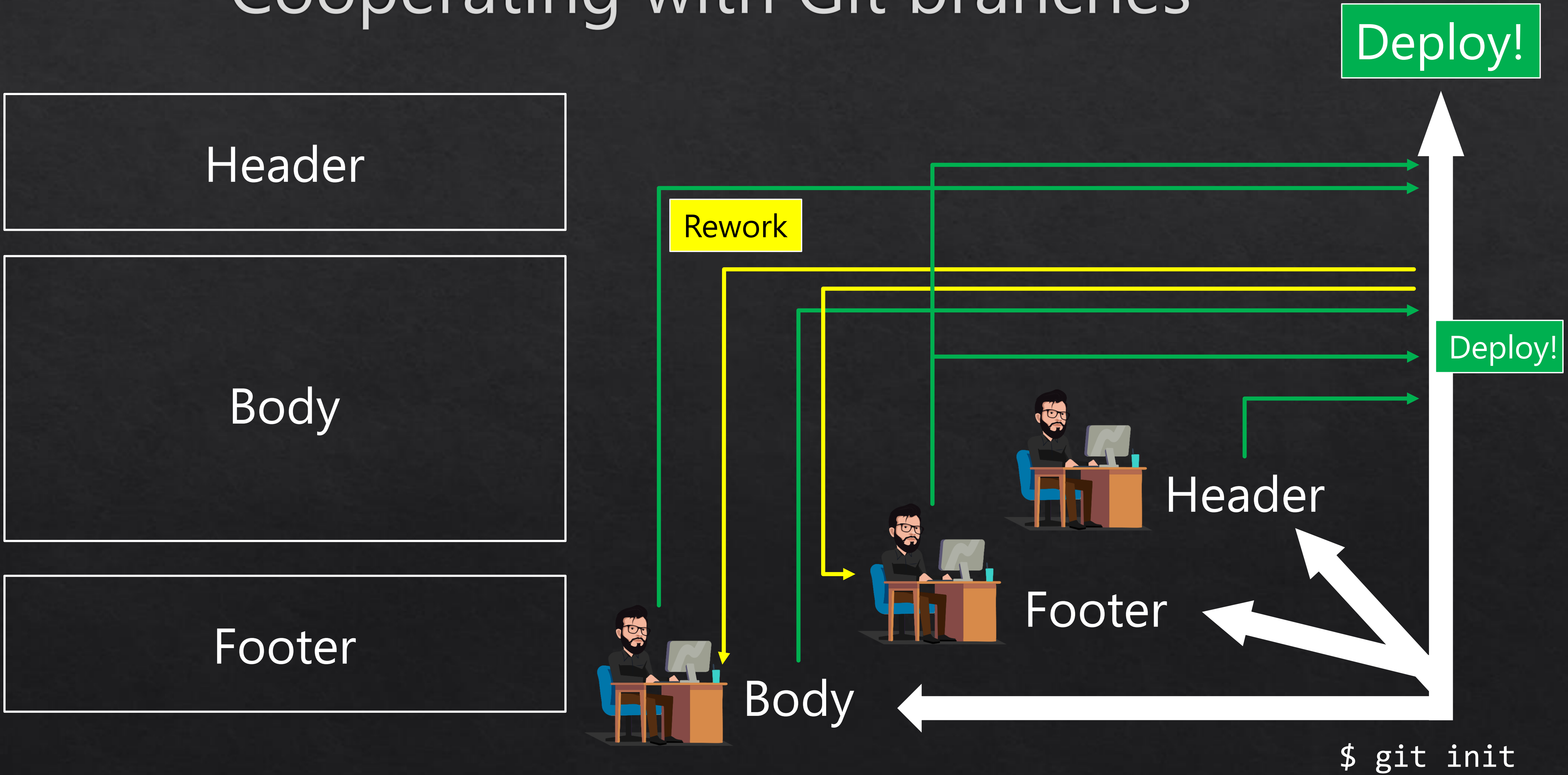


Image source: https://www.vervelogic.com/images/img/hire-developer/ishank-sir.png

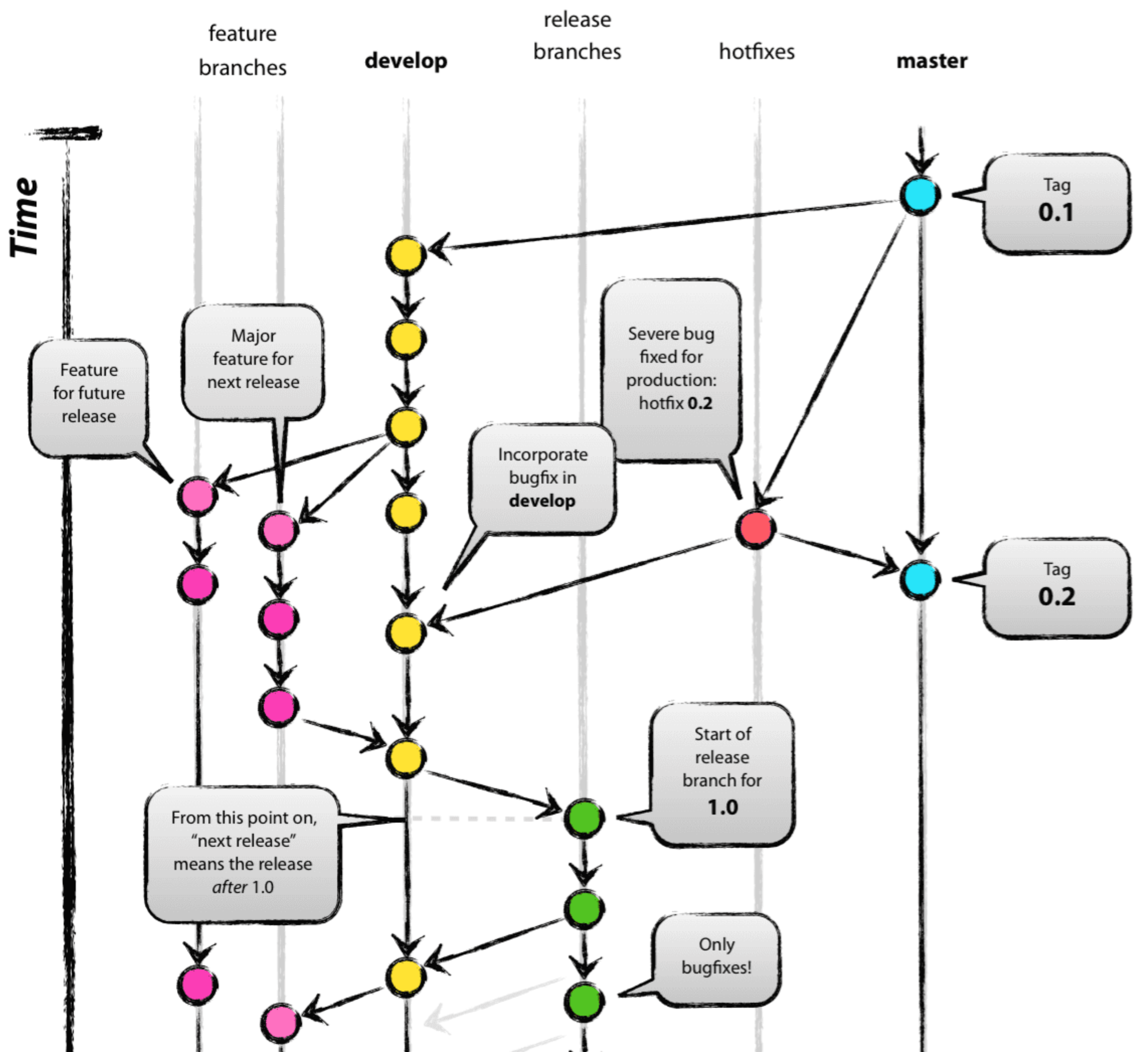
Wrong version was deployed!

Cooperating with Git branches



Cooperate with Git branches

- Always keep code in **`master`** branch is available.
- Use branches to differ different stages of code.
- Use **`Pull Requests`** to allow others to evaluate your contribution.
- **`Merge`** code after testing.


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Severe bug fixed for production: hotfix 0.2
Major feature for next release
Feature for future release
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!

Local & Remote Branches

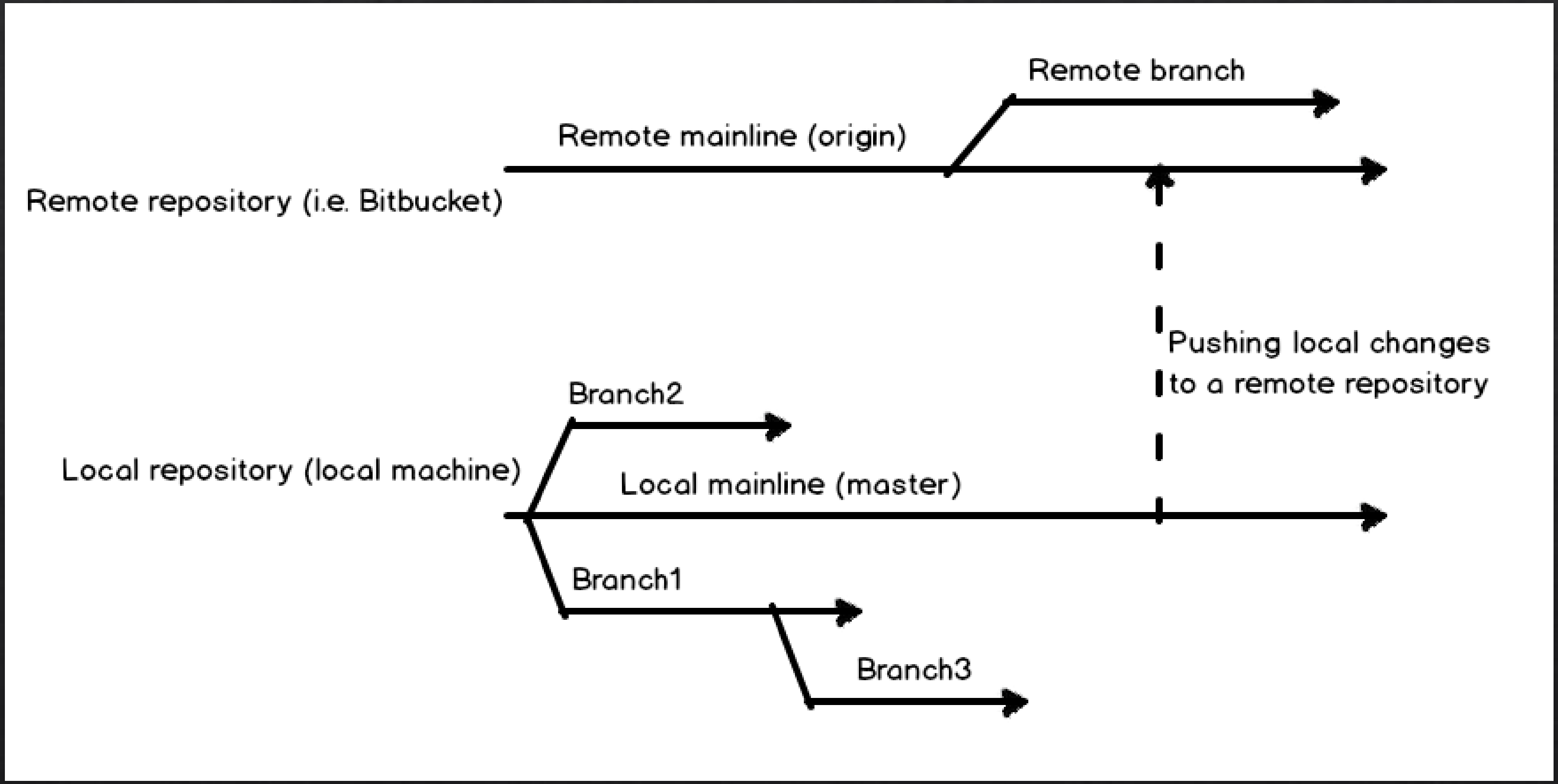

Remote branch
Remote mainline (origin)
Remote repository (i.e. Bitbucket)
Pushing local changes to a remote repository
Branch2
Local repository (local machine)
Local mainline (master)
Branch1
Branch3

Image source: https://solutioncenter.apexsql.com/wp-content/uploads/2016/05/word-image-78.png

Frequently Asked Questions

- What is GitHub? What is its relationship with Git?
  - GitHub is an online Git repository hosing service.
- Why Git requires user to save manually?
  - Unlike documents, software code has a long lifecycle. It is necessary to push only meaningful code.
- Why there is conflict in Git?
  - Because there is situation that Git cannot handle and only human can.
- Why there is a fast-forward problem?
  - Because you didn’t pull before push.

Useful Tips

- Always create repository with a **`.gitignore`** file.
- Always **`pull`** before modify files.
- Always **`commit`** with **meaningful** information.
- Try to make use of **branches** by **`forking`**.
- Try to work with git by **command line**.
- Be social in git communities!

References & Useful Resources

- GitHub Guides: https://guides.github.com/
- Git Cheat sheet: https://services.github.com/on-demand/resources/cheatsheets/
- GitHub Help: https://help.github.com/
- The Book *Pro Git*: https://git-scm.com/book/en/v2
- Handling Conflicts: https://stackoverflow.com/questions/161813/how-to-resolve-merge-conflicts-in-git