Version Control System

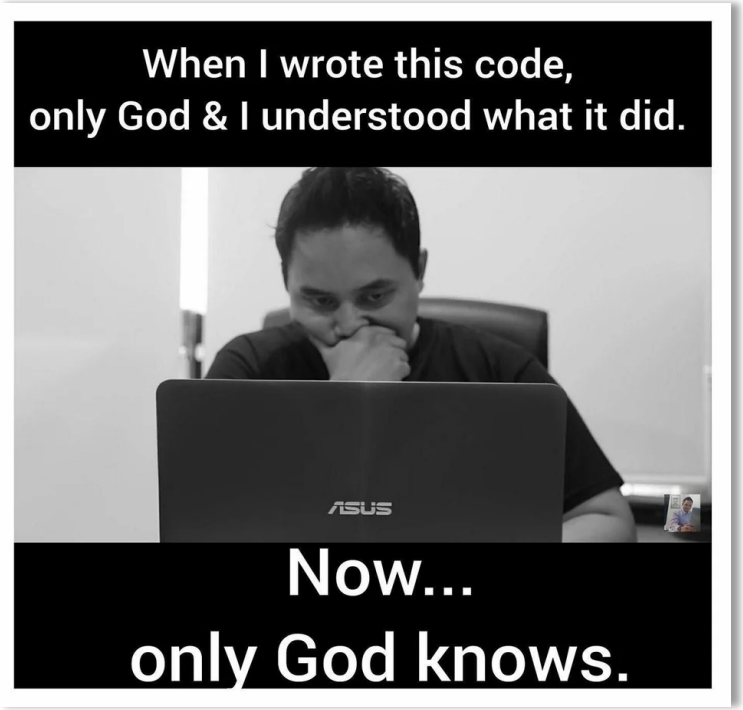
Version Control System

What?

- Version Control = Revision Control = Source Control = Source Code Management
- Managing changes to documents, computer programs, large websites, and other collections of information
- Revision numbers: letters or numbers used to represent each change

Why?

- Keep tracking changes
- Revert to a specific checking point
- Work with people



Version Control System - Git

- Version Control Systems
  - Revision Control System (RCS) ***- Local***
  - Subversion (SVN) – *e.g.,* Google Code (2006 – 2016) ***– Centralized***
  - Concurrent Versions System (CVS) – Grandpa of VCS ***– Centralized***
  - Git – most popular VCS, > 80% market share ***– Distributed***
  - Mercurial – much simpler than Git ***– Distributed***
- Google Drive / OneDrive is **NOT** a VCS!
- What is Git?

  Git is a distributed (decentralized) version-control system for tracking changes in source code during software development.
- Git service providers
  - Azure DevOps
  - Bitbucket (We will use Bitbucket in this workshop)
  - Github
  - Gitlab

Git: Installation

- macOS (use Homebrew)

  $ brew update

  $ brew install git

- Linux (Ubuntu)

  $ sudo apt-get update

  $ sudo apt-get install git

- Windows (Git client, WSL, Git GUI client …)

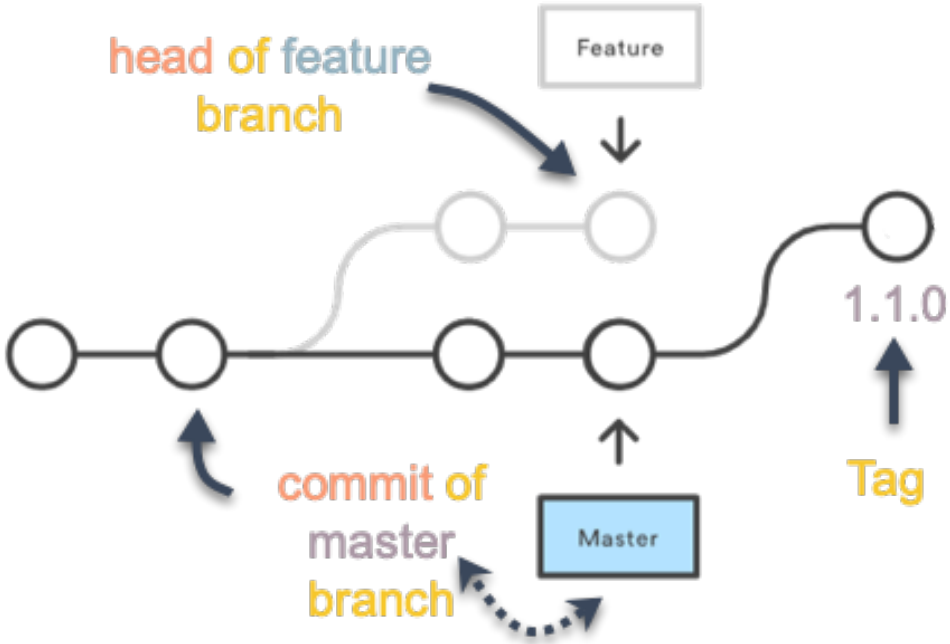
  Google is your best friend ;p

  Not sure how? Click here.

  Powered by lmgtfy

Git Glossary

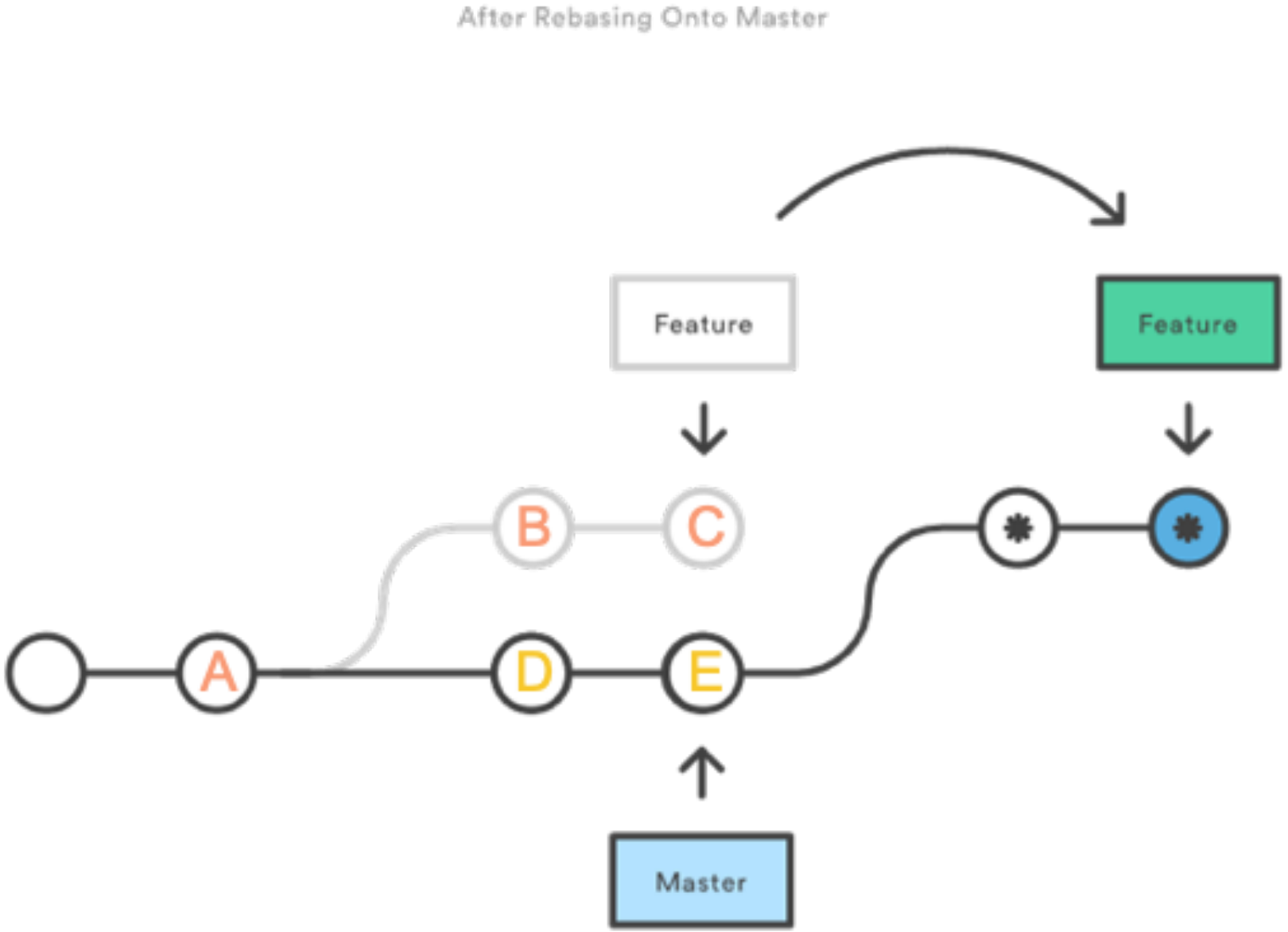
- **Repository**: a collection of commits, branches and tags etc.
- **Commit**: an individual change to a repository.
- **Branch**: represents an independent line of development. It is a parallel version of a repository.
- **head**: A named reference to the commit at the tip of a branch.
- **HEAD**: The current branch.
- **Tag:** A reference typically used to mark a particular point in the commit chain.



Git Glossary

- **Remote repository:** A version of something that is hosted on a server (Bitbucket, Github, Gitlab).
- **Clone:** creating a repository from another repository
- **Checkout**: create a local working copy from the repository, or to switch between the branches
- **Pull**: Fetching changes and merging them.
- **Push:** Sending your changes to a remote repository.
- **Merge:** take changes from one branch into another.
- **Rebase:** To reapply a series of changes from one branch to another and reset the head of that branch to the result.

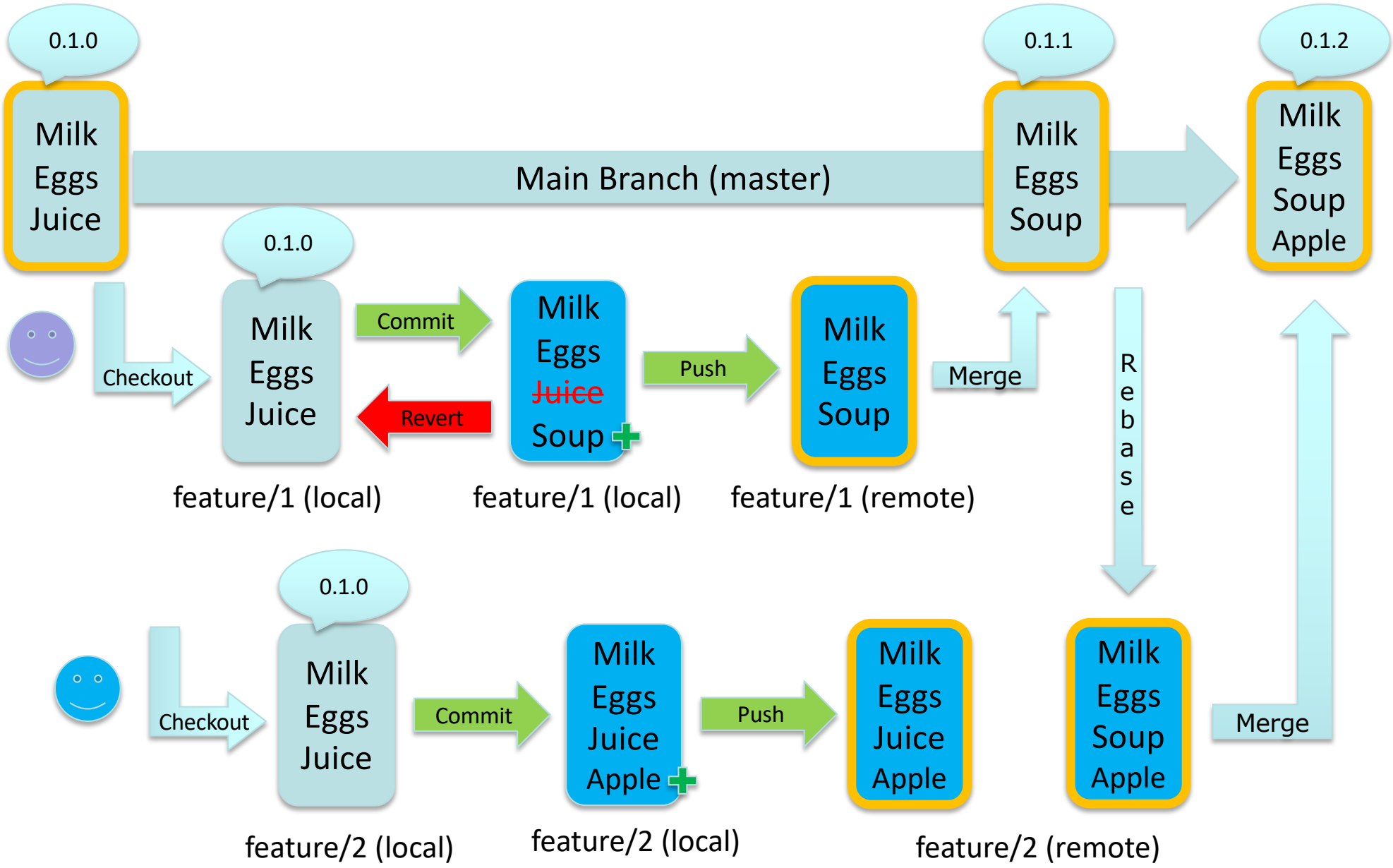
Rebase



Git: a simple guide

- Branch naming conventions
  - Main branch: e.g. master
  - Development branch: e.g. develop
  - Other branches:
    - **Feature branch**: add a new feature, e.g. feature/add-a-new-feature
    - **Bugfix branch**: fix a bug, e.g. bugfix/fix-a-bug
    - **Release branch**: make a release, e.g. release/1.0
    - **Hotfix branch**: hotfix a bug in a release, e.g. hotfix/fix-another-bug

Git at a glance



Git: a simple guide

- Register an account on Bitbucket, Github or Gitlab
  - Bitbucket - https://bitbucket.org
  - Github - https://github.com/
  - Gitlab - https://gitlab.com

- Create a Git repository (Bitbucket)
- Checkout a new repository

$ git clone https://<username>@<hostname>/path/to/repo.git

Git: a simple guide

- Branching
  - Create a develop branch (from GUI)
  - Create two feature branches

    $ git checkout -b feature/demo-1

    $ git checkout -b feature/demo-1-1
  - Switch between branches

    $ git checkout feature/demo-1
  - List all branches

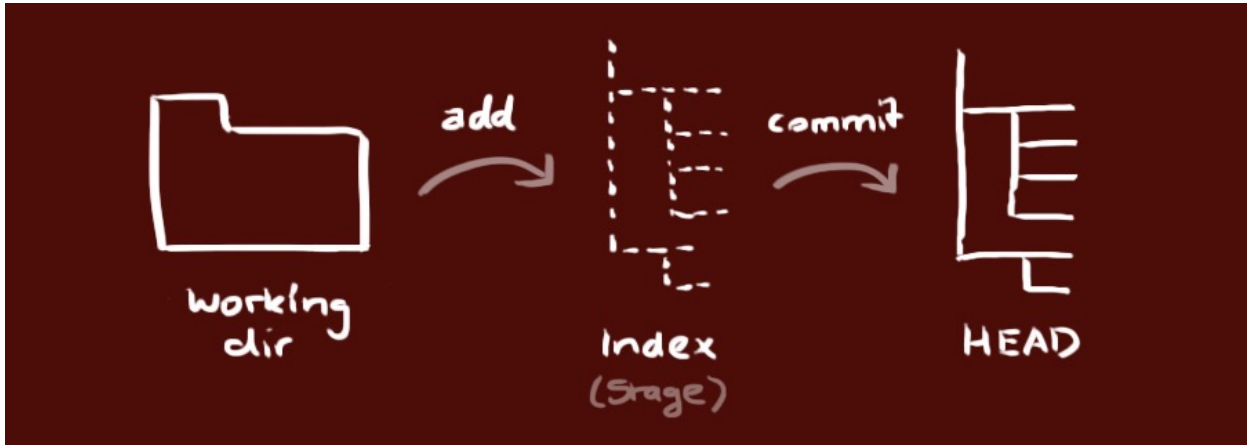
    $ git branch
  - Delete a branch

    $ git branch -d feature/demo-1-1
  - Push a local branch to remote

    $ git push origin feature/demo-1

Git: a simple guide

- Simple Workflow



  - Made changes
  - Add

    $ git add <filename>
  - Commit

    $ git commit -m “commit message”
  - Push

    $ git push origin feature/demo-1

    $ git push --set-upstream origin feature/demo-1

Git: a simple guide

- Pull Request (Merge Request) – Code review
  - Create Pull Request for others to review your code
  - Approve or Decline a Pull Request
  - Merge a Pull Request when all reviewers approved

- Fetch and pull the changes from remote
  - Fetch updates

    $ git fetch -a
  - Pull changes from remote

    $ git pull

Git: a simple guide

- Conflicts
  - Conflict may happen if users changed the same code block
  - Code cannot be merged until conflicts are resolved
  - Scenario
    - User A checked out *develop* branch and changed one line in README.md on feature branch *feature/demo-2*

```
10   10
11        - You'll start by editing this README.md file to learn how to edit a file in Bitbucket.
     11   + You'll start by editing this README.txt file to learn how to edit a file in Bitbucket.
12   12
```

    - At the same time User B checked out *develop* branch and changed the same line in README.md on feature branch *feature/demo-3*

```
10   10
11        - You'll start by editing this README.md file to learn how to edit a file in Bitbucket.
     11   + You'll start by editing this README-txt file to learn how to edit a file in Bitbucket.
12   12
```

    - User A merged the change to *develop* branch
    - User B created a Pull Request to merge *feature/demo-3* to *develop* branch and saw conflict

Git: a simple guide

- Conflicts

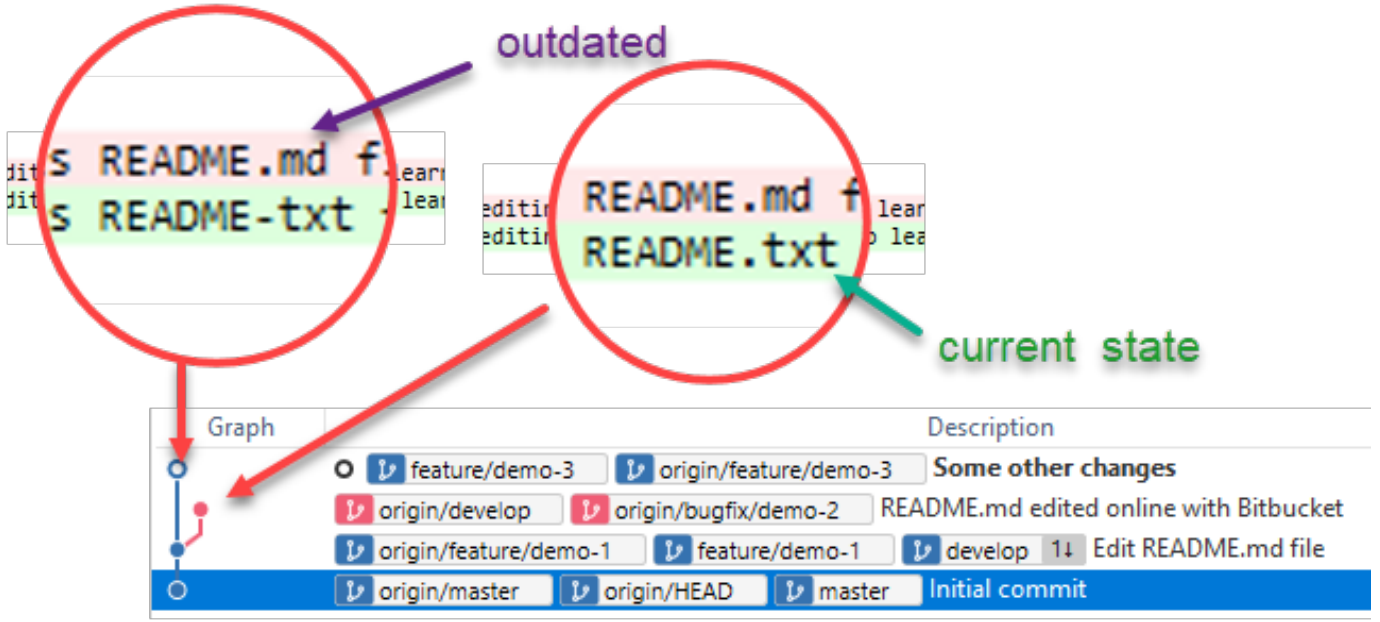
README.md CONFLICT

Side-by-side diff | View file | Comment | •••

Conflict: File modified in both source and destination

To be able to merge you will need to resolve the conflicts manually. Learn how.

```
...
 8   8
 9   9  ## Edit a file
10  10
    11 +<<<<<<< destination:8f0f6df83f494f0803ca8b92425dd9902285b4eb
11  12  You'll start by editing this README.txt file to learn how to edit a file in Bitbucket.
    13 +=======
    14 +You'll start by editing this README-txt file to learn how to edit a file in Bitbucket.
    15 +>>>>>>> source:d7f6d93ff63c16106dce51b645d6054eb30eba0e
12  16
```



Git: a simple guide

- Rebase and resolve the conflict (use tool)

$ git fetch -a

$ git pull --rebase origin develop

```
## Edit a file
```

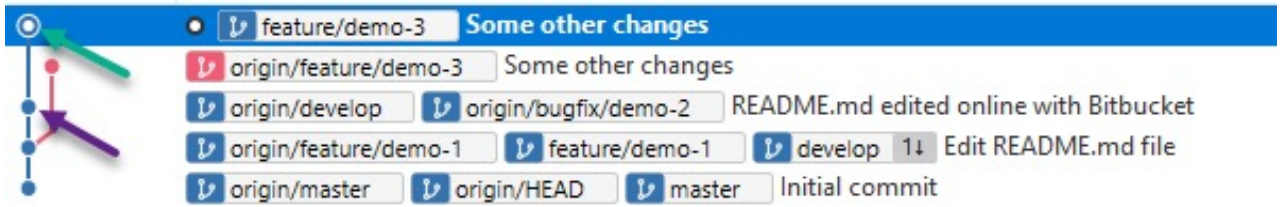
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes

```
<<<<<<< HEAD (Current Change)
You'll start by editing this README.txt file to learn how to edit a file in Bitbucket.
=======
You'll start by editing this README-txt file to learn how to edit a file in Bitbucket.
>>>>>>> Some other changes (Incoming Change)

1. Click **Source** on the left side.
```

$ git add READM.md

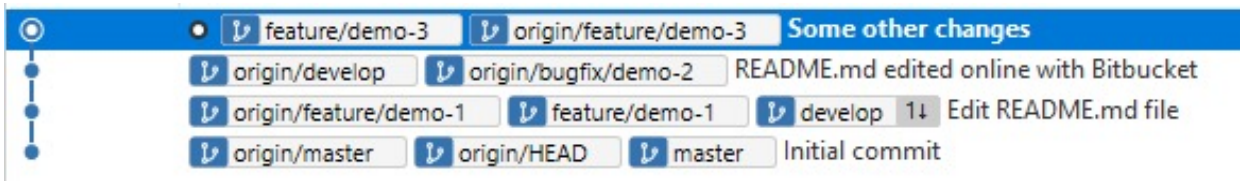
$ git rebase --continue



Git: a simple guide

- Rebase and resolve the conflict (use tool)

$ git push -f origin feature/demo-3



README.md MODIFIED — Side-by-side diff | View file | Comment | •••

```
...
 8   8
 9   9  ## Edit a file
10  10
11      -You'll start by editing this README.txt file to learn how to edit a file in Bitbucket.
    11  +You'll start by editing this README-txt file to learn how to edit a file in Bitbucket.
12  12
13  13  1. Click **Source** on the left side.
14  14  2. Click the README.md link from the list of files.
...
```

References

- Version Control, https://en.wikipedia.org/wiki/Version_control
- Git, https://en.wikipedia.org/wiki/Git
- A simple git guide, http://rogerdudler.github.io/git-guide/
- Git terminology, https://git-scm.com/docs/gitglossary