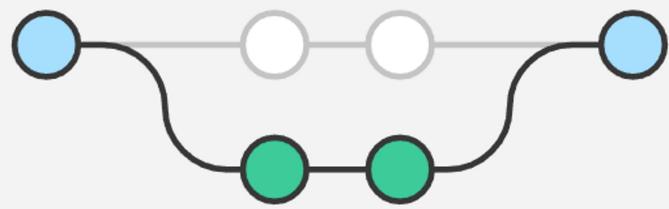


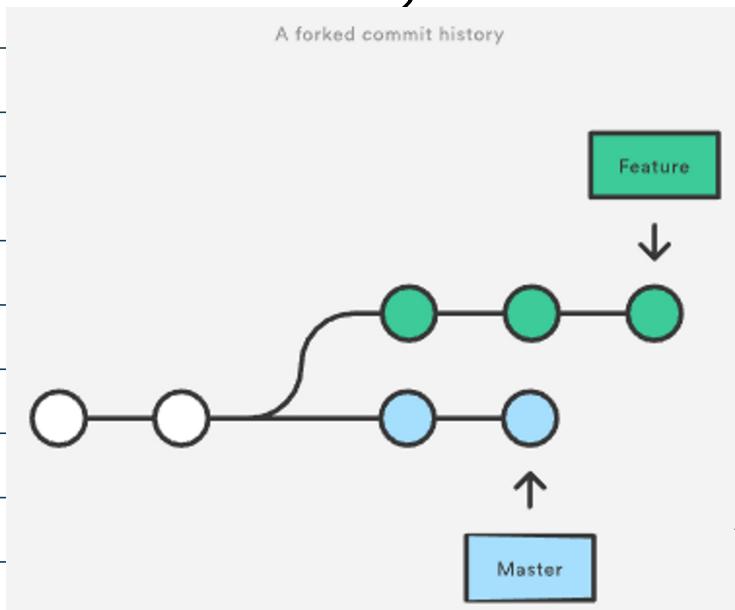
Merging VS. Rebasing



Overview

git rebase solves the same problem as git merge
both commands are designed to integrate changes from one branch into another branch

Example: you're working on a new feature in a dedicated branch, and then master updates:



forked history

- new commits are relevant to the feature you're working on
- to incorporate changes into your feature branch, either rebase || merge

Merge Option

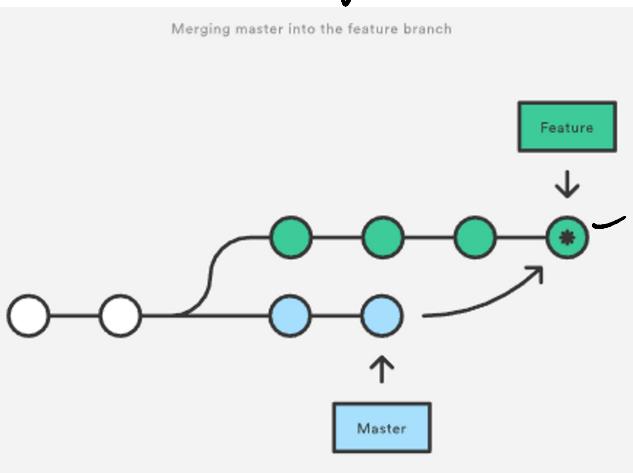
```
git checkout feature
```

merge master branch into feature branch

```
git checkout feature  
git merge master
```

merge master branch into feature branch

- a new "merge commit" is created in feature branch



merge commit

existing branches are not changed
in any way

feature branch will have extraneous merge commit every time you need to incorporate upstream changes

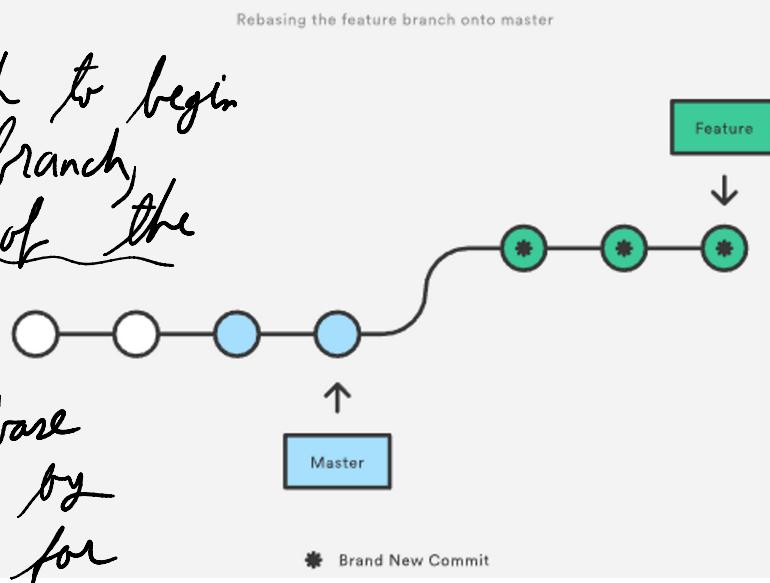
- if active master

Rebase Option

```
git checkout feature  
git rebase master
```

you can rebase the feature branch onto master branch

moves entire feature branch to begin on the tip of the master branch, effectively incorporating all of the new commits in master



instead of merge commit, rebase re-writes the project history by creating brand new commits for each commit in original branch

each commit in original branch

Create New Commit

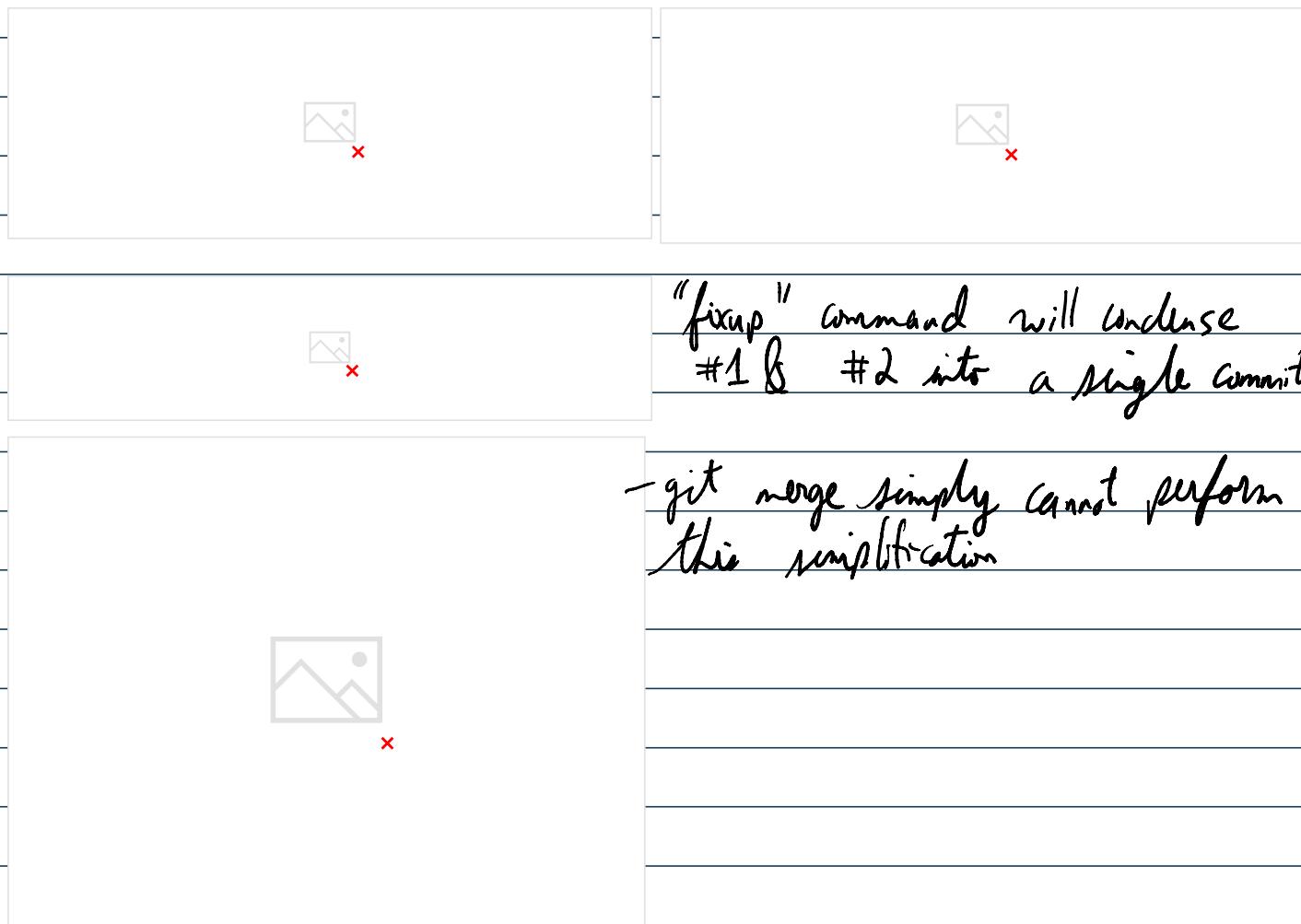
+ much cleaner project history

+ perfectly linear → w/out forks

- easier to navigate w/ git log, git bisection, gitk

Interactive Rebasing

- alter commits as they are moved to the new branch



The golden Rule of Rebasing

- must learn when not to do it

- must learn when not to do it
- never to use it on public branches



rebase moves all commits in master onto tip of feature

Force-pushing

pushing rebased master branch to a remote repo



Workflow Walkthrough

- create dedicated branch for each feature



Local Cleanup

- 2 options for new base:
 - ① master branch
 - ② an earlier commit in your feature



↳ interactive rebase of only the last 3 commits

Interactive rebase of only the last 3 commits



x