

SWPP

2019.03.08

- Git
- Branch
- Merge
- Squash
- GitHub

Git

- A version control system (software)

Git

- A version control system (software)

```
foo.c void foo() {  
      ...  
      }
```

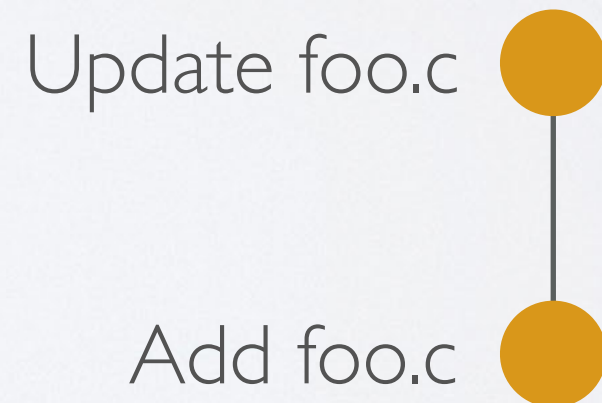
Add foo.c 

Git

- A version control system (software)

foo.c

```
int foo() {  
    ...  
}
```



Git

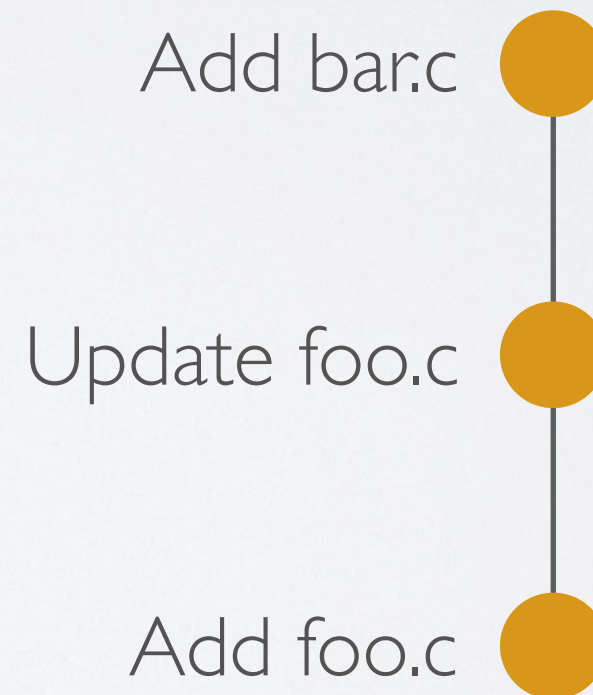
- A version control system (software)

foo.c

```
int foo() {  
    ...  
}
```

bar.c

```
int bar() {  
    ...  
}
```



Git

- A version control system (software)

foo.c

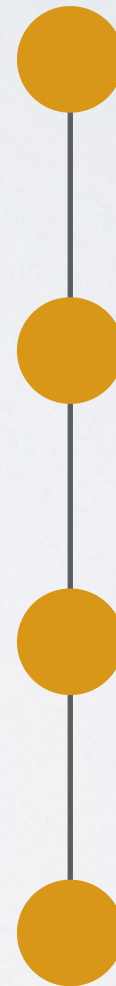
```
int foo() {  
    ...  
}
```

Remove bar.c

Add bar.c

Update foo.c

Add foo.c



Git

Working Directory

Staging Area

Repository

Git

`git init`



Working Directory

Staging Area

Repository

Git

`git init`



Working Directory

`git add`



Staging Area

Repository

Git

`git init`



Working Directory

`git add`

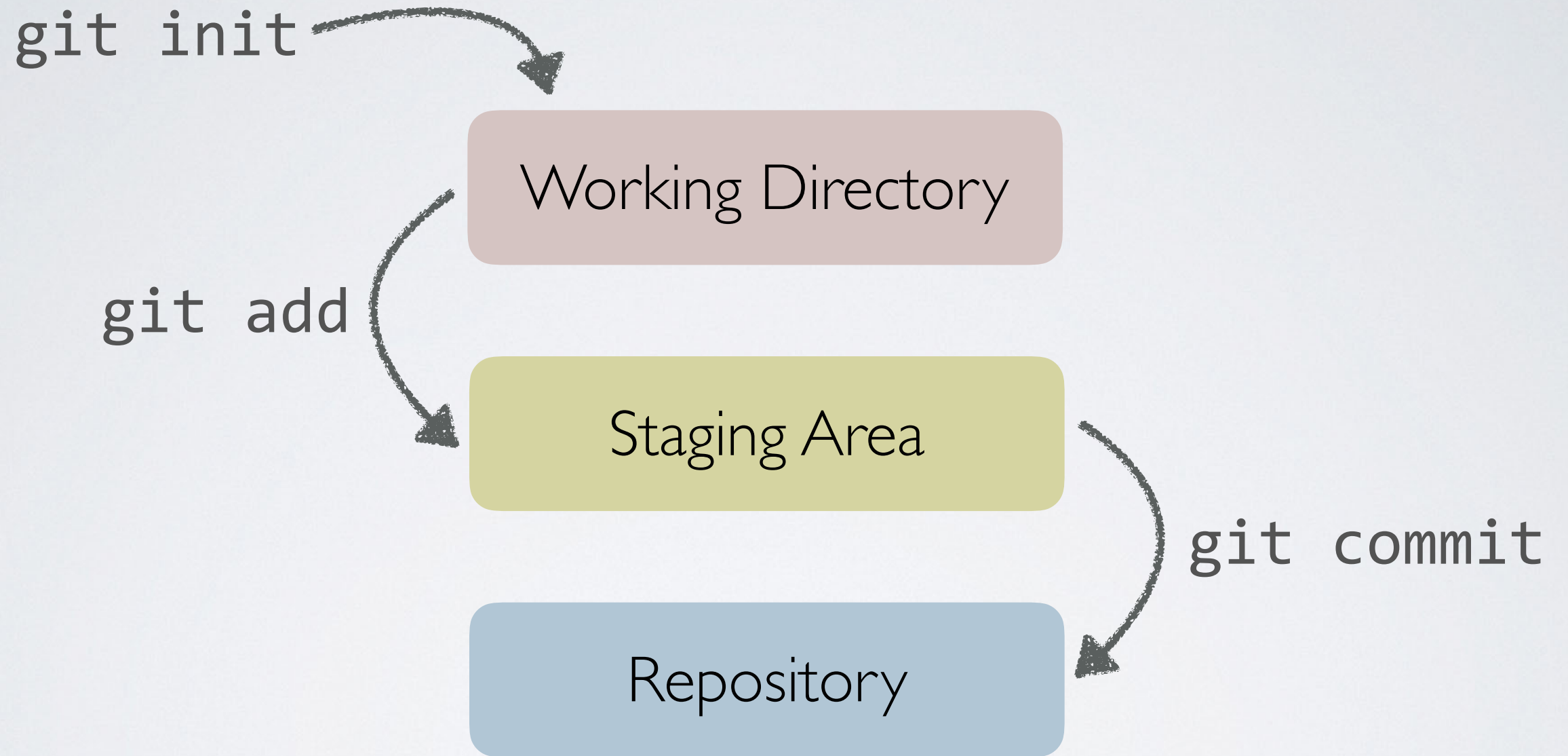


Staging Area

`git commit`



Repository



Git

- **git init**
 - initializes a new git repository
- **git add**
 - stages a change
- **git commit**
 - saves the snapshot
- **git status**
 - shows the status of changes
- **git log**
 - shows commit log (history)

Git

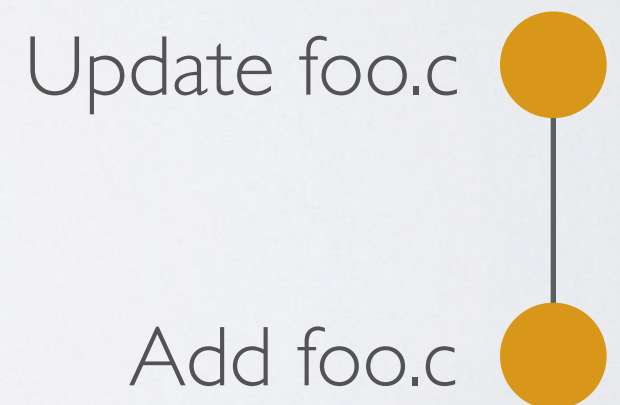
- **git init**
 - initializes a new git repository
- **git add**
 - stages a change
- **git commit**
 - saves the snapshot
- **git status**
 - shows the status of changes
- **git log**
 - shows commit log (history)

Add foo.c



Git

- **git init**
 - initializes a new git repository
- **git add**
 - stages a change
- **git commit**
 - saves the snapshot
- **git status**
 - shows the status of changes
- **git log**
 - shows commit log (history)



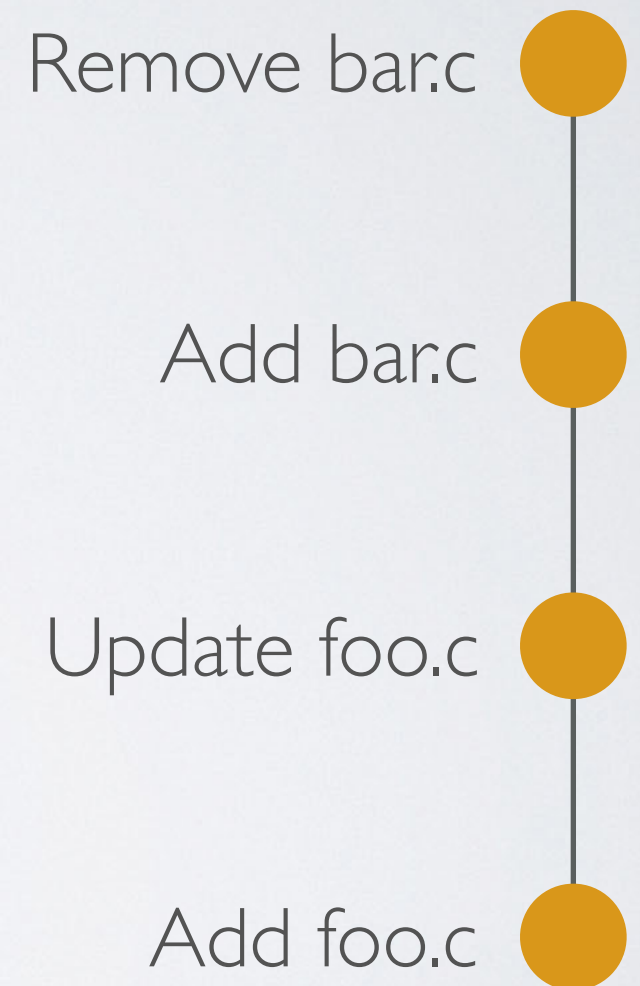
Git

- **git init**
 - initializes a new git repository
- **git add**
 - stages a change
- **git commit**
 - saves the snapshot
- **git status**
 - shows the status of changes
- **git log**
 - shows commit log (history)

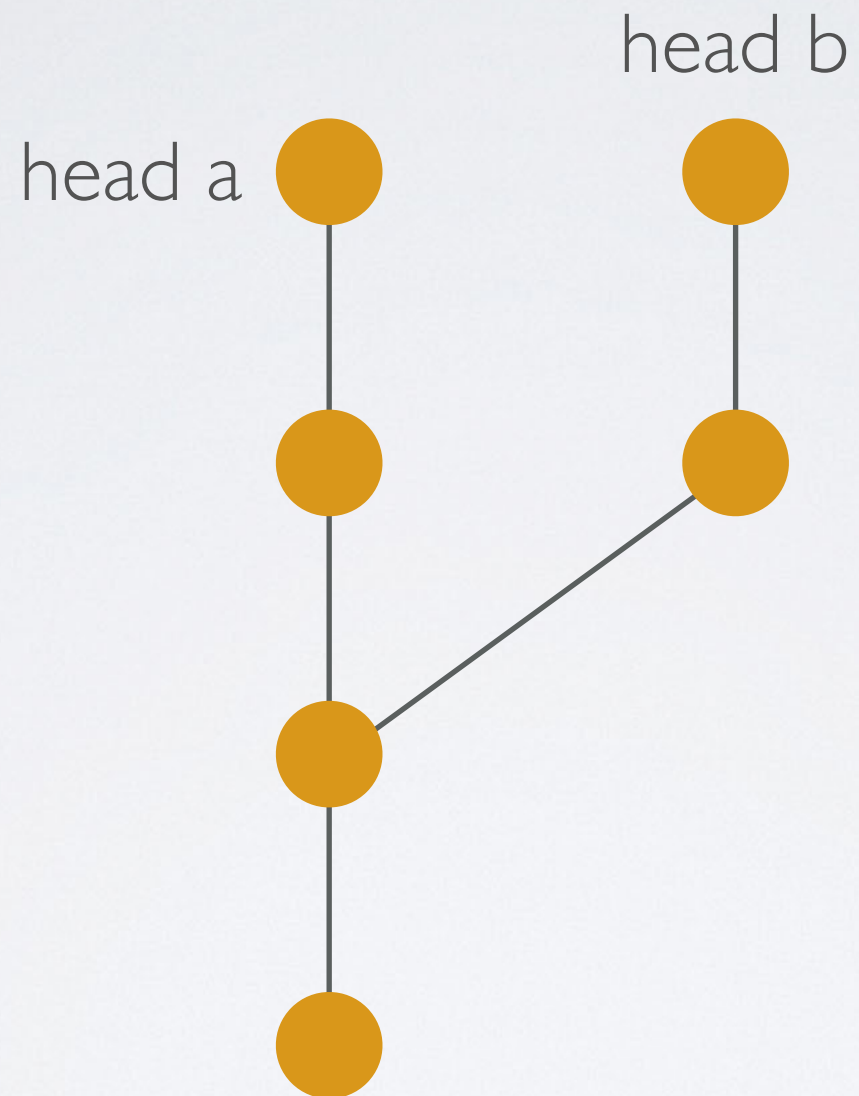


Git

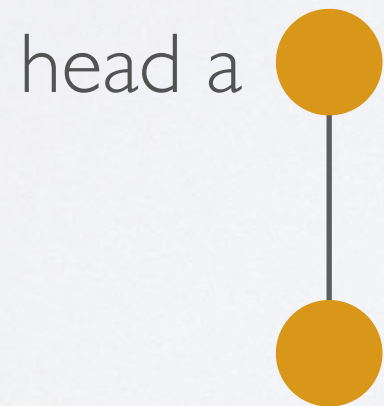
- **git init**
 - initializes a new git repository
- **git add**
 - stages a change
- **git commit**
 - saves the snapshot
- **git status**
 - shows the status of changes
- **git log**
 - shows commit log (history)



Branch

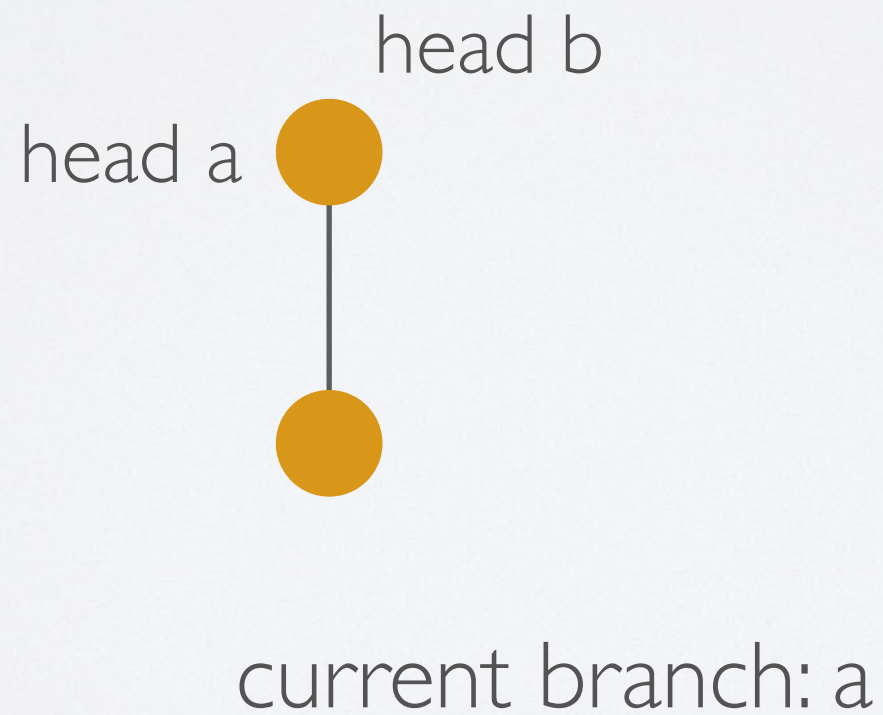


Branch



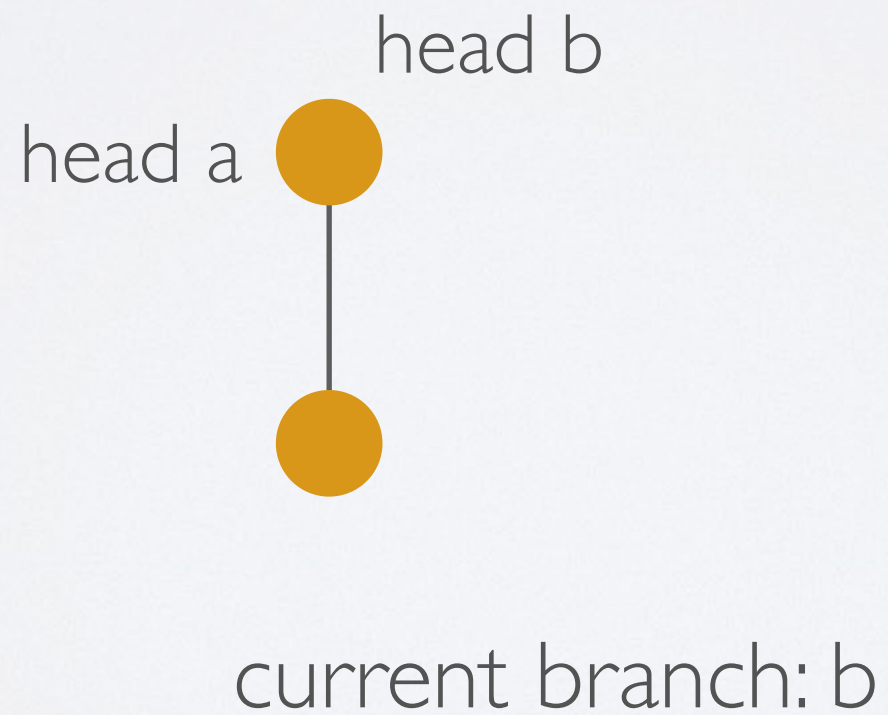
Branch

`git branch b`



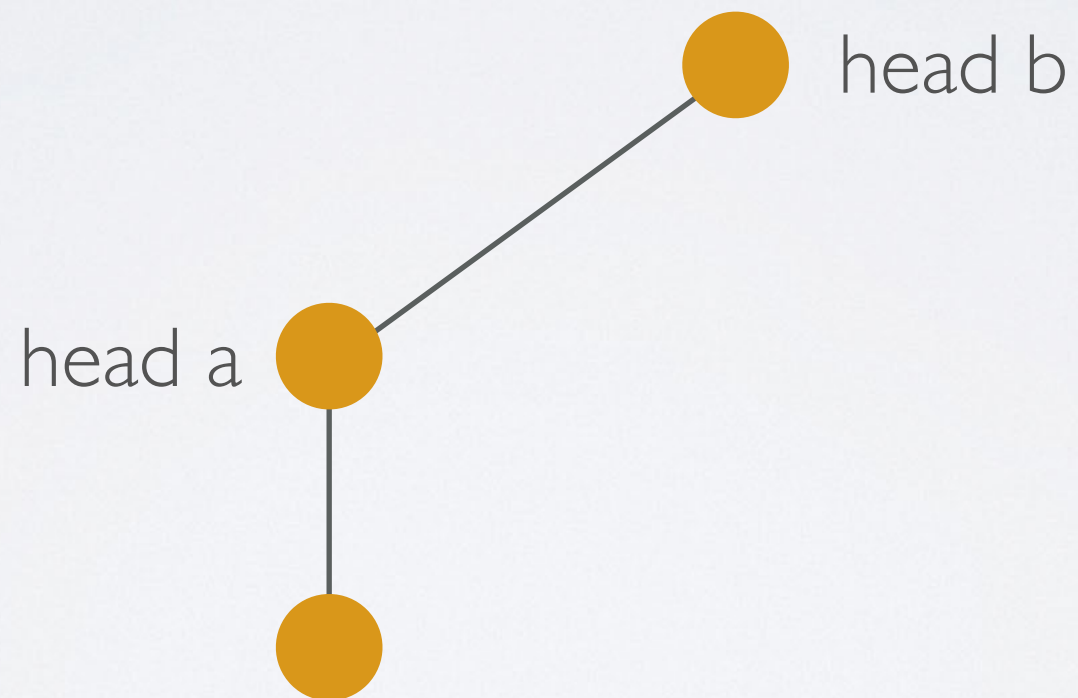
Branch

`git checkout b`



Branch

`git commit`



current branch: b

Branch

head b

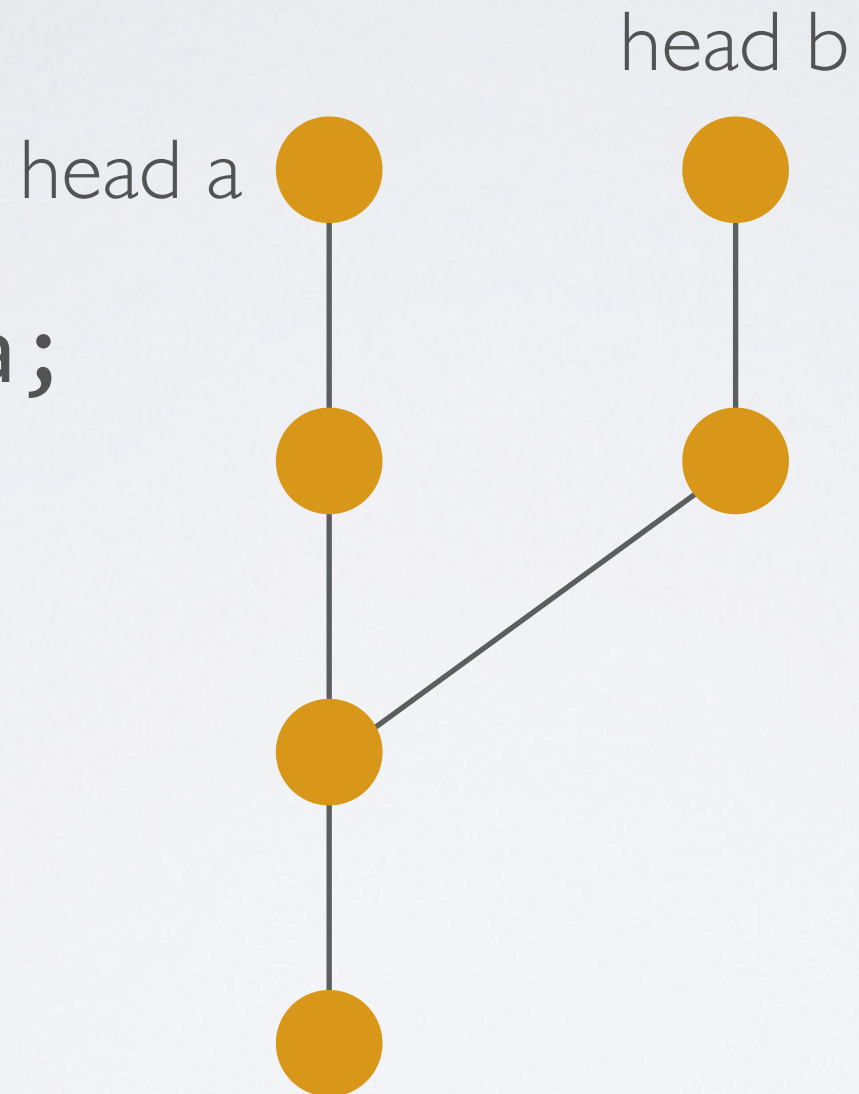
```
git commit
```

head a

current branch: b

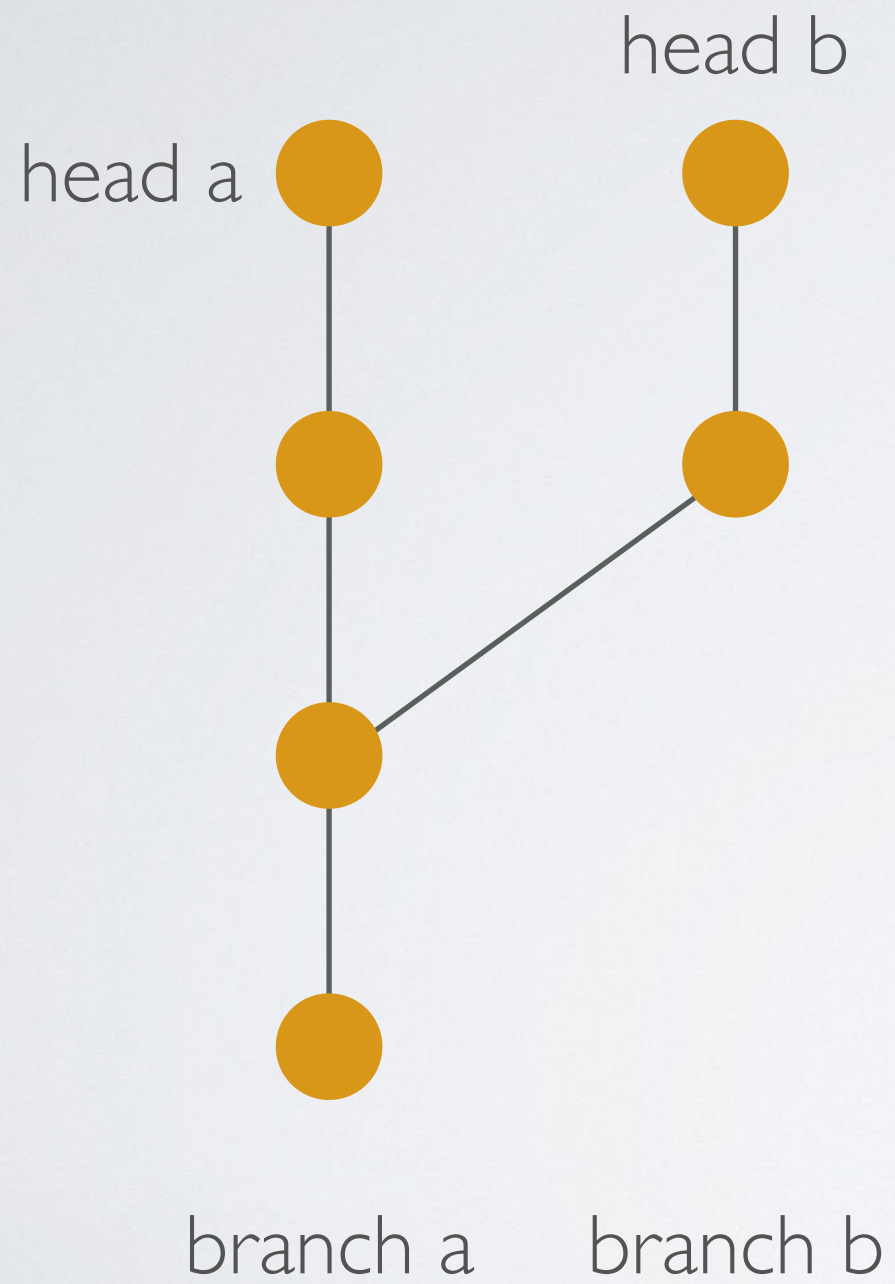
Branch

```
git checkout a;  
git commit;  
git commit
```

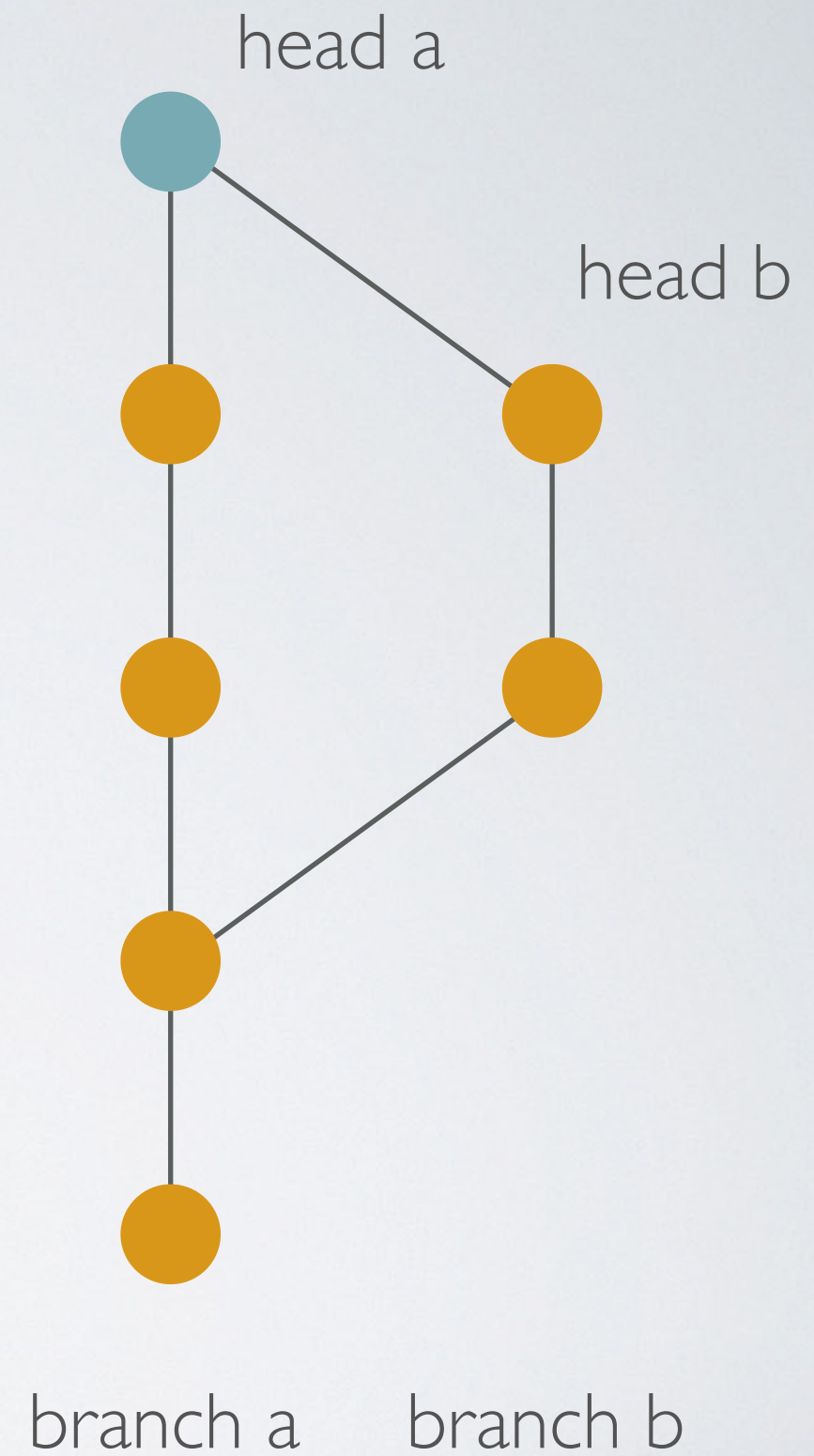
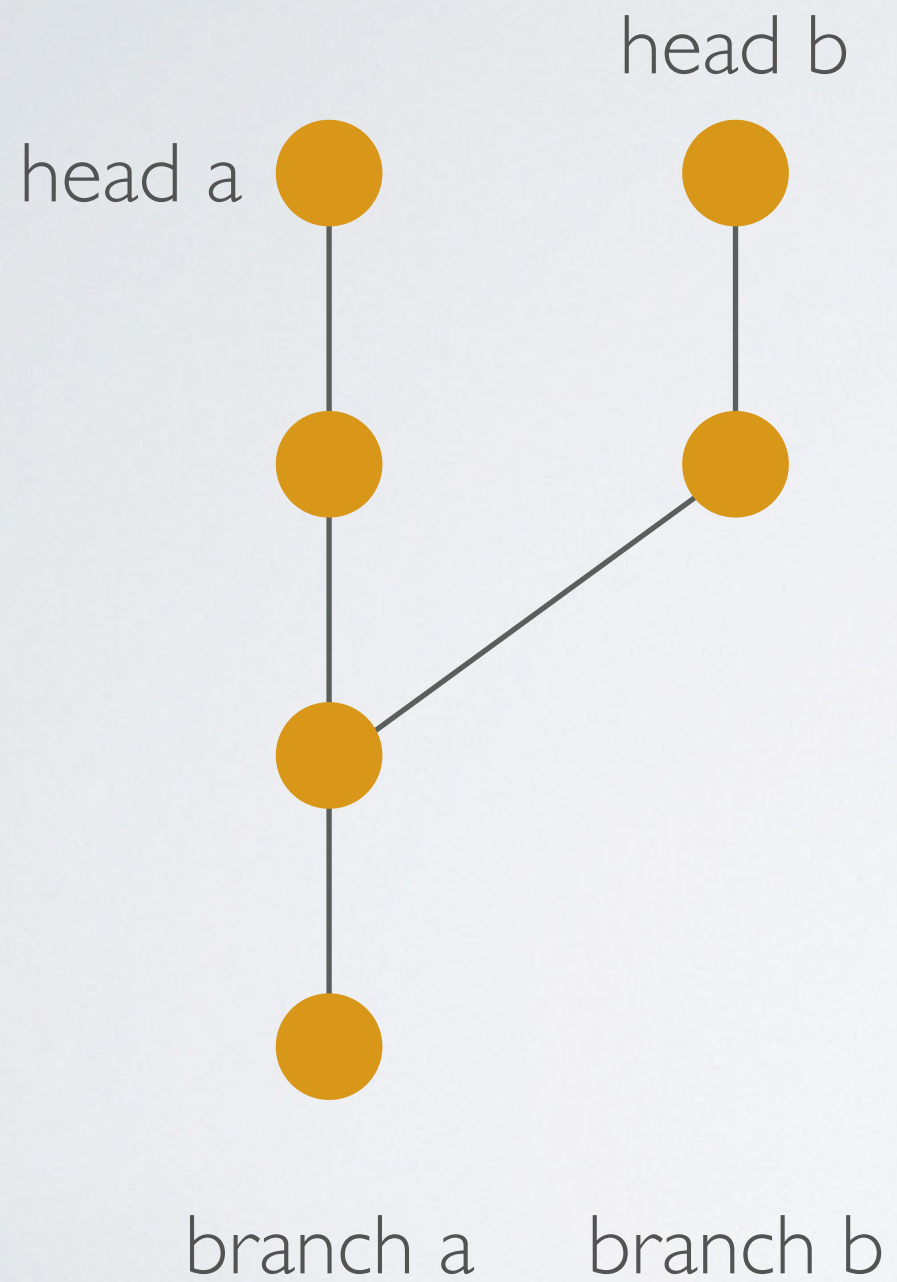


current branch: a

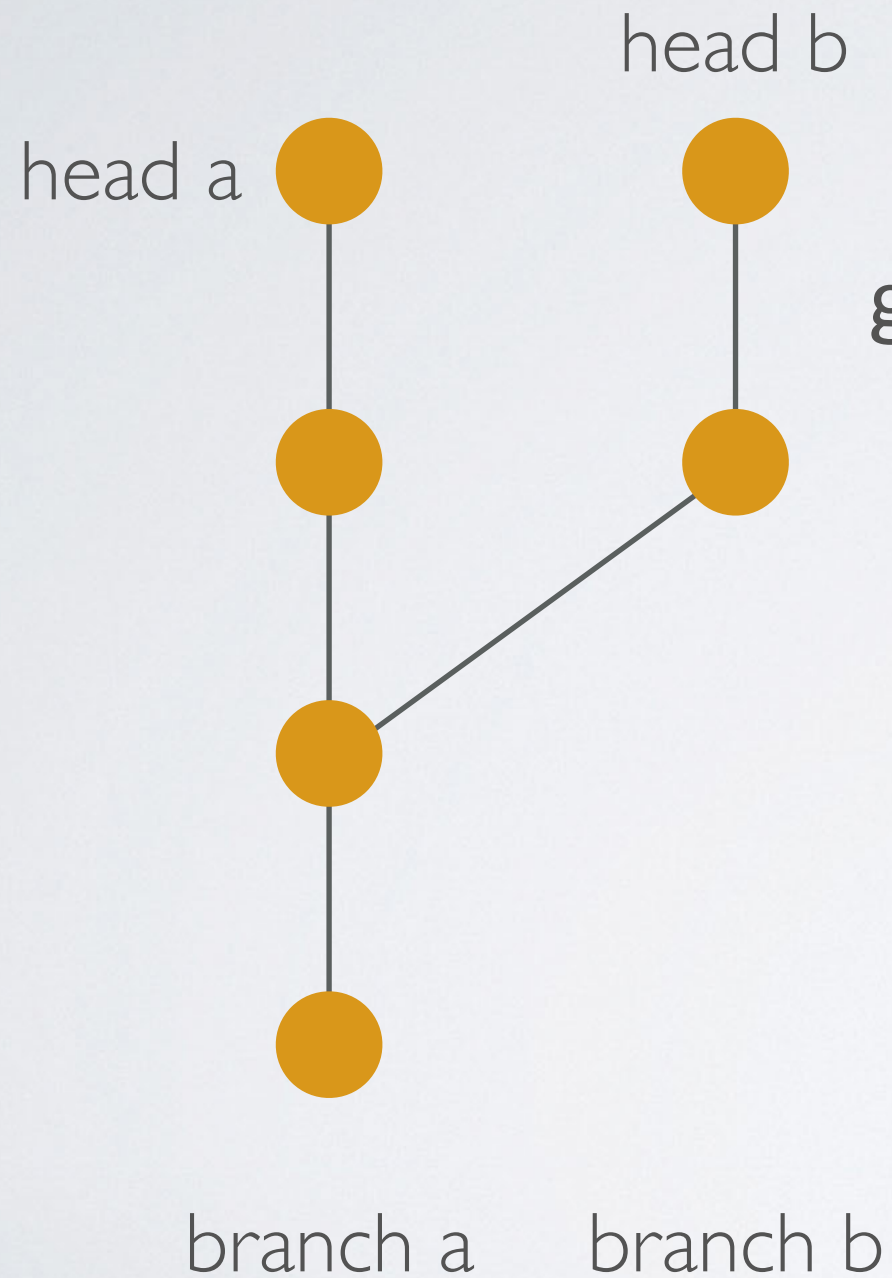
Merge



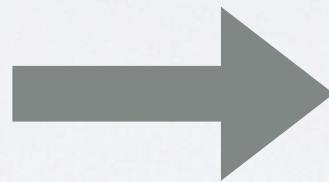
Merge



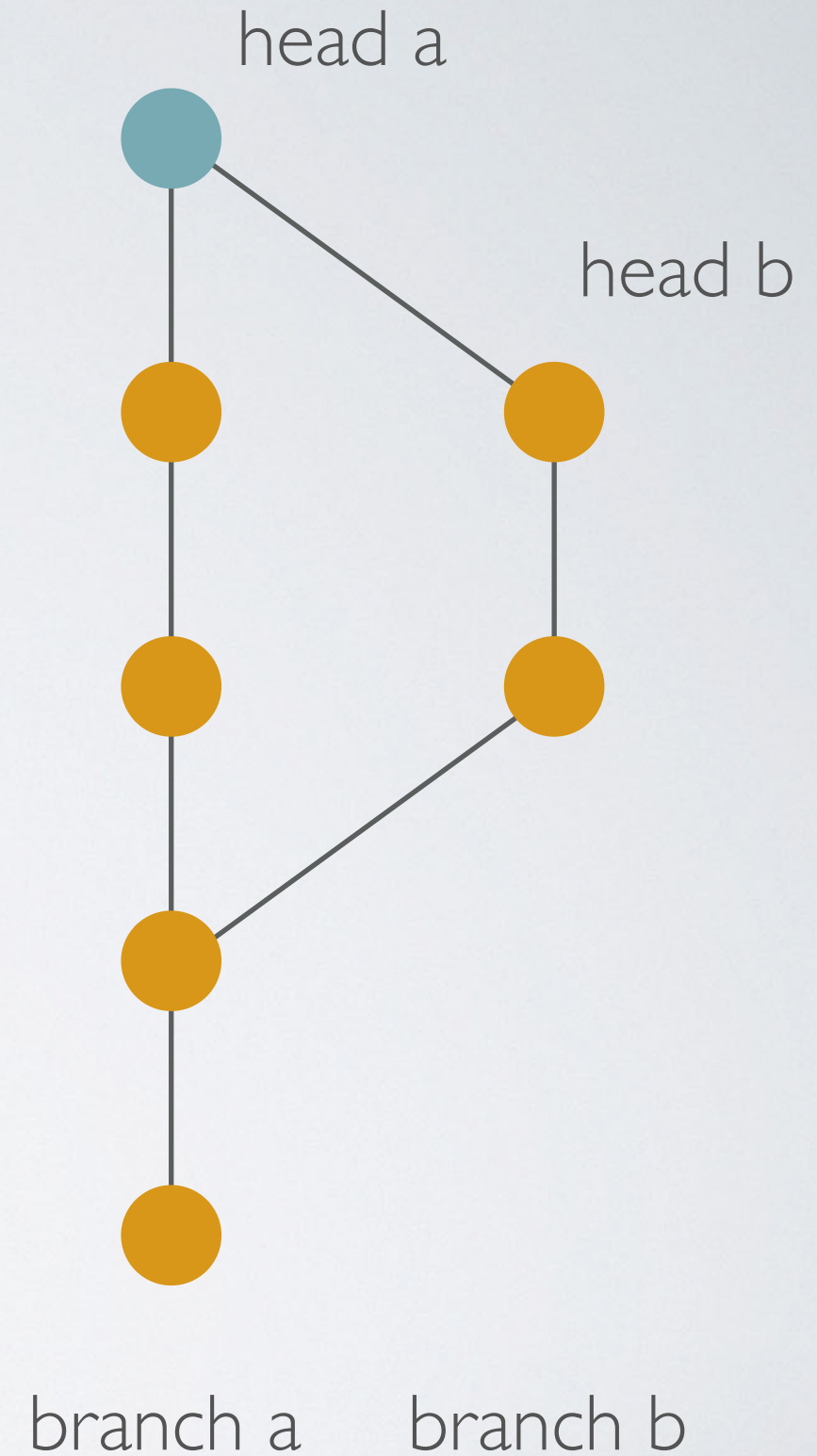
Merge



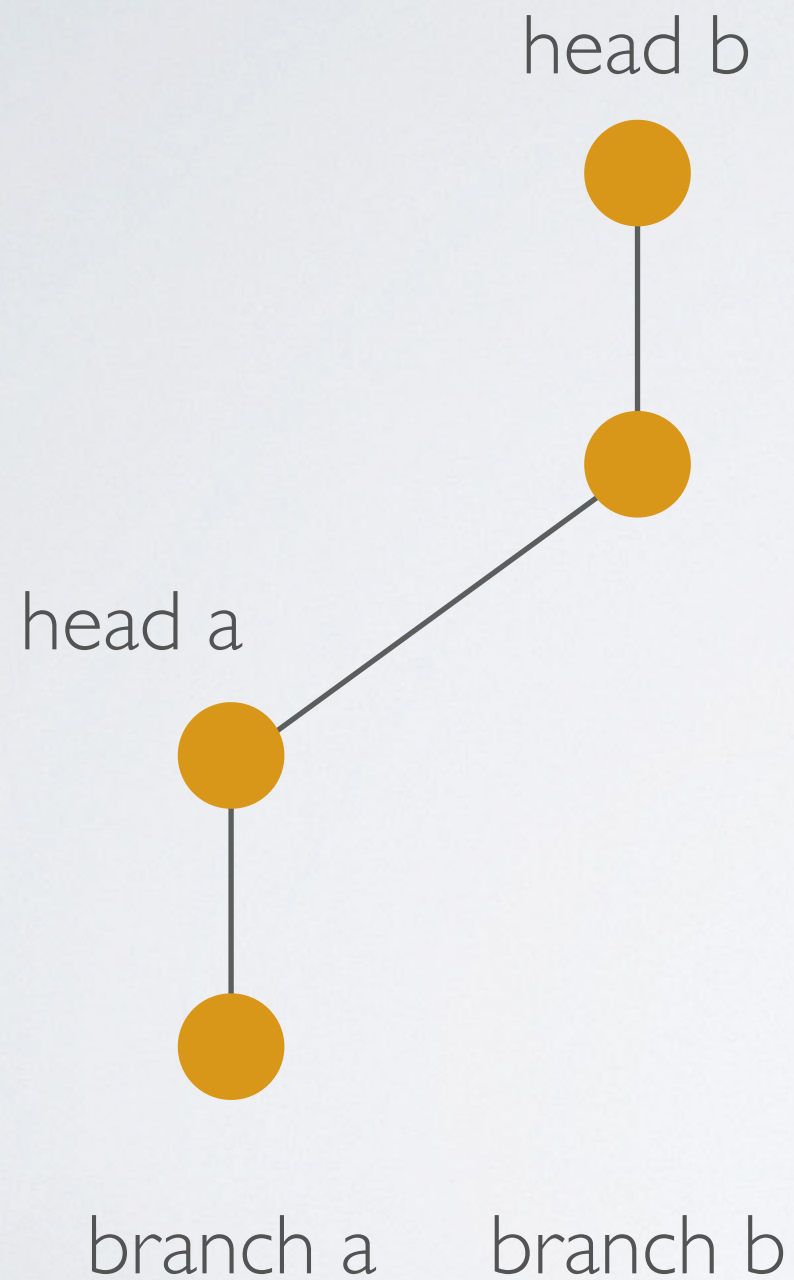
`git checkout a;`
`git merge b`



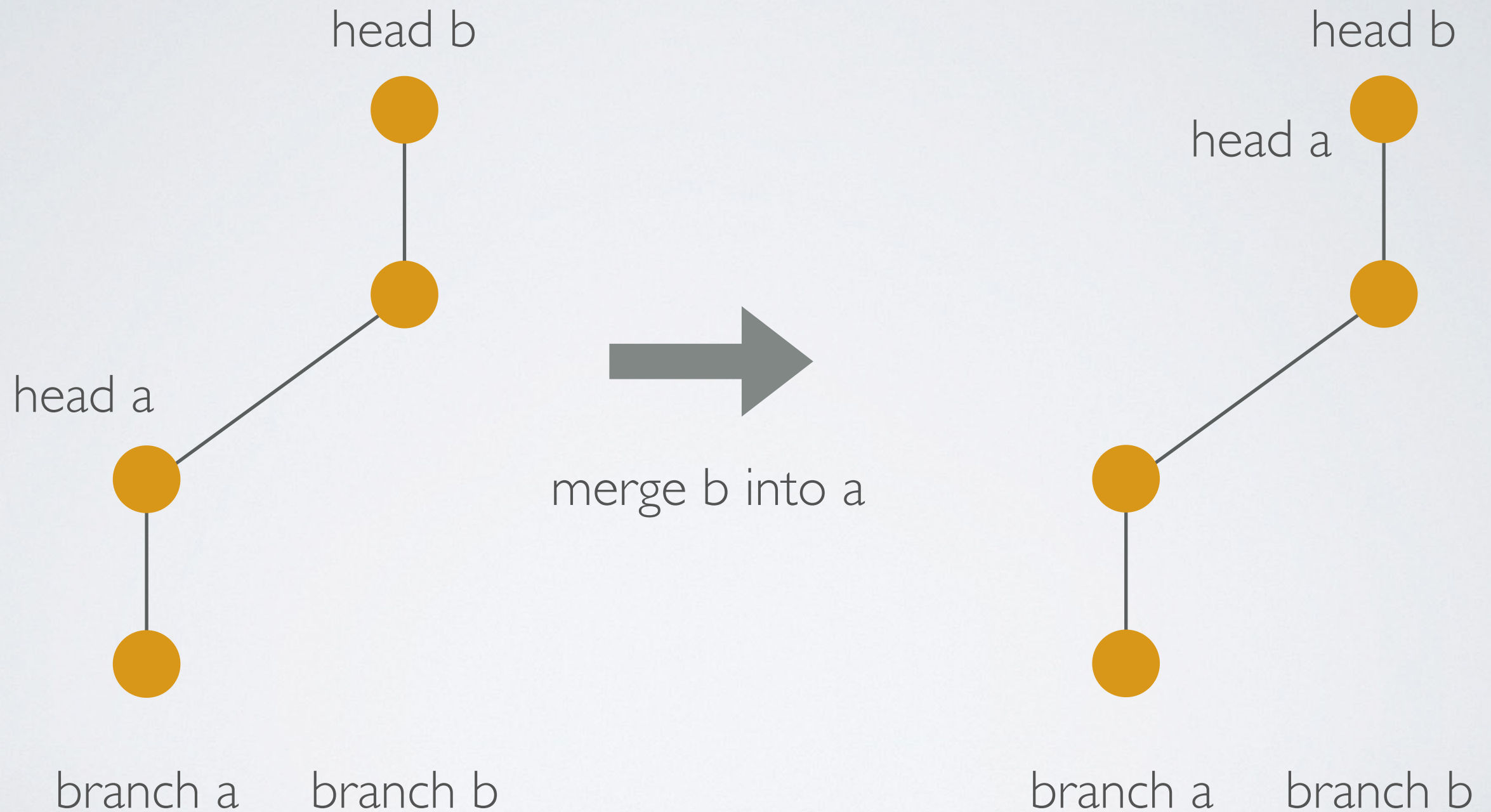
merge b into a



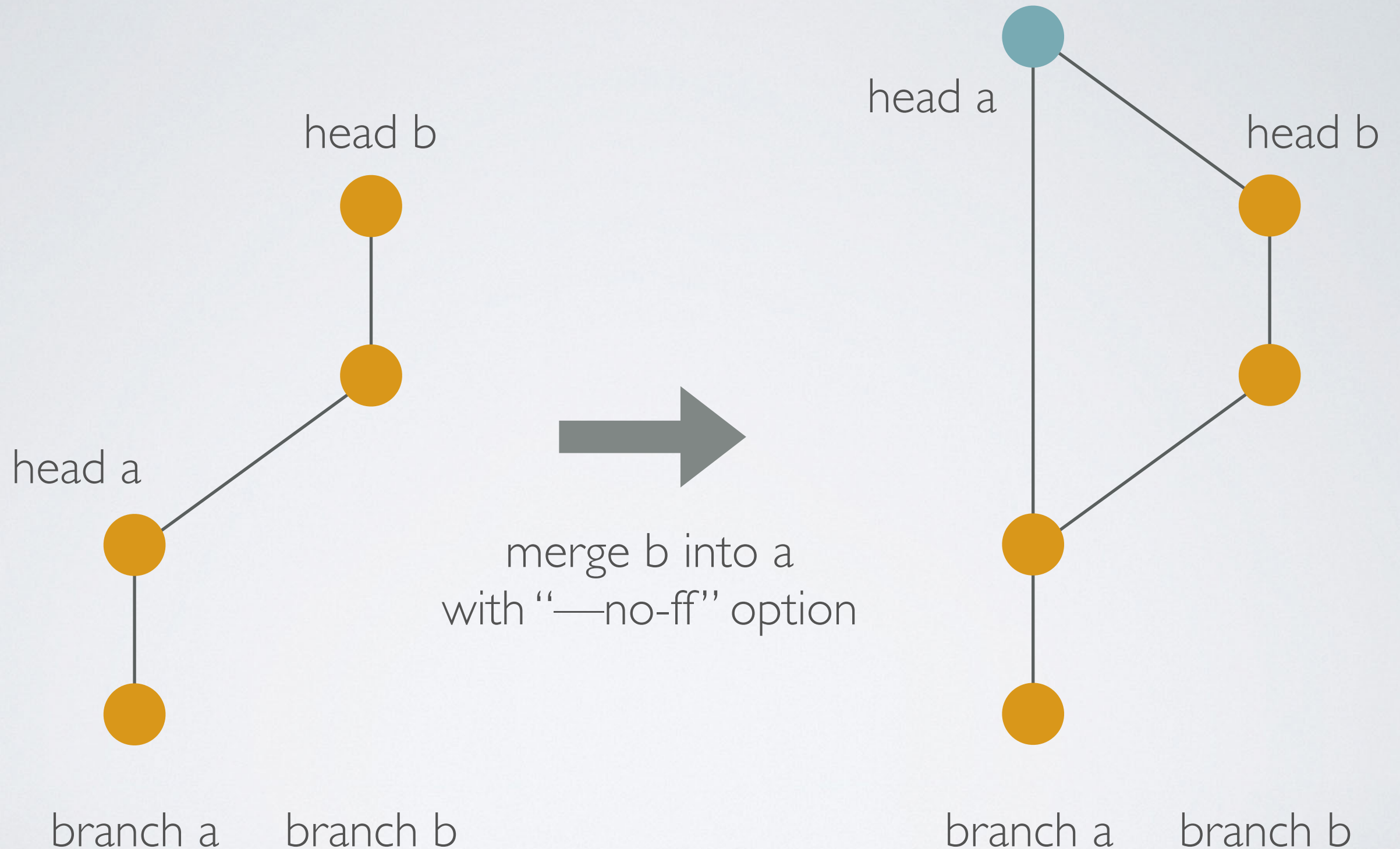
Fast-Forward Merge



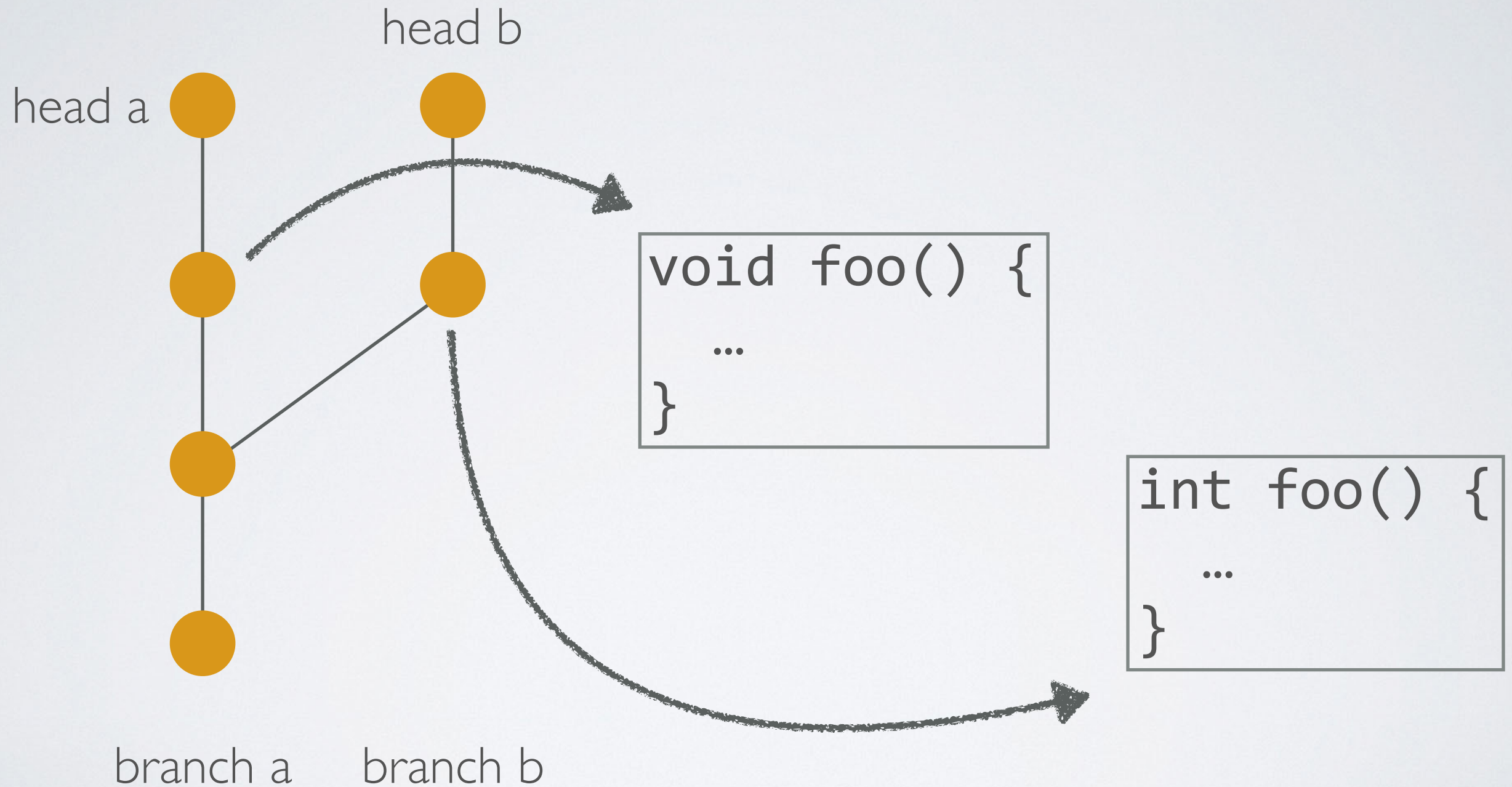
Fast-Forward Merge



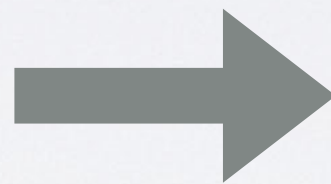
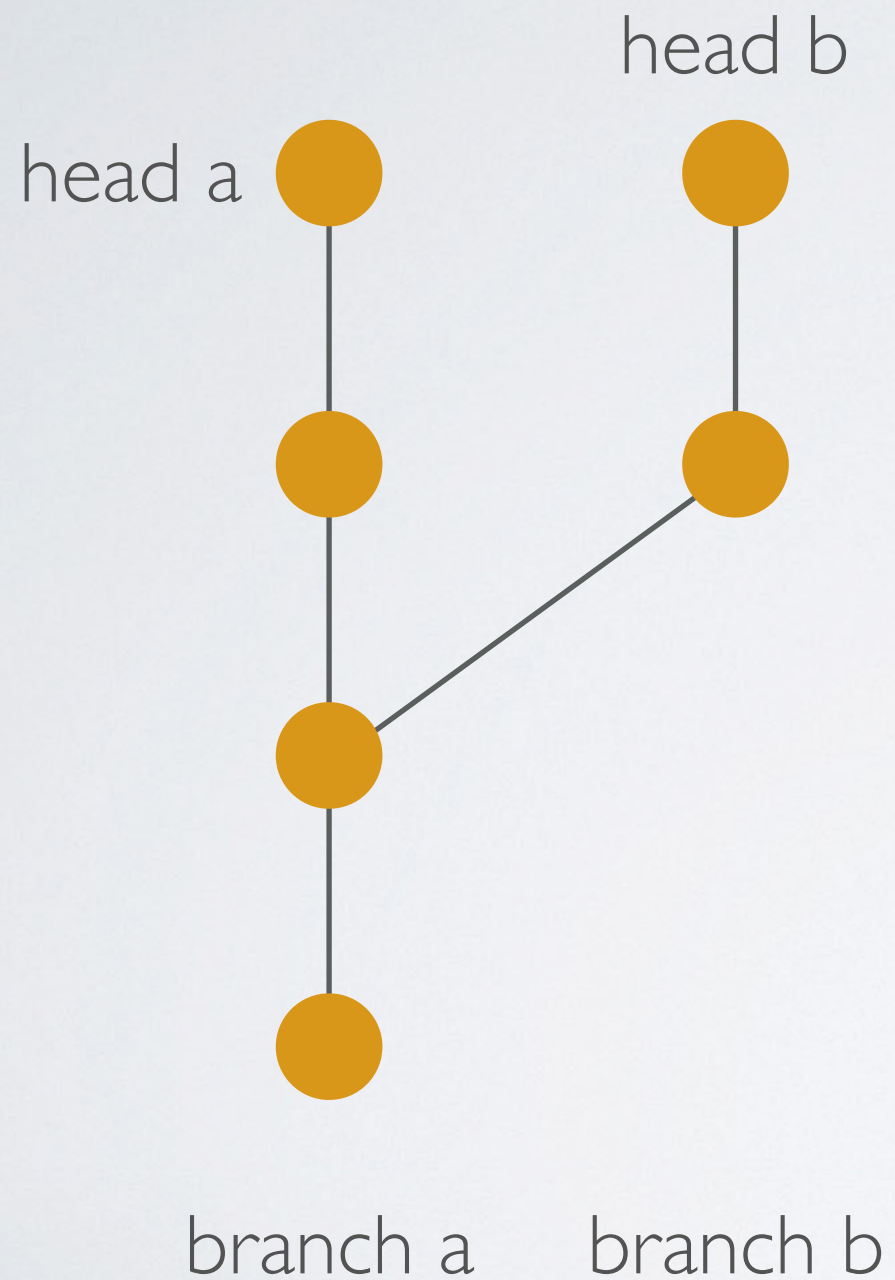
Non Fast-Forward Merge



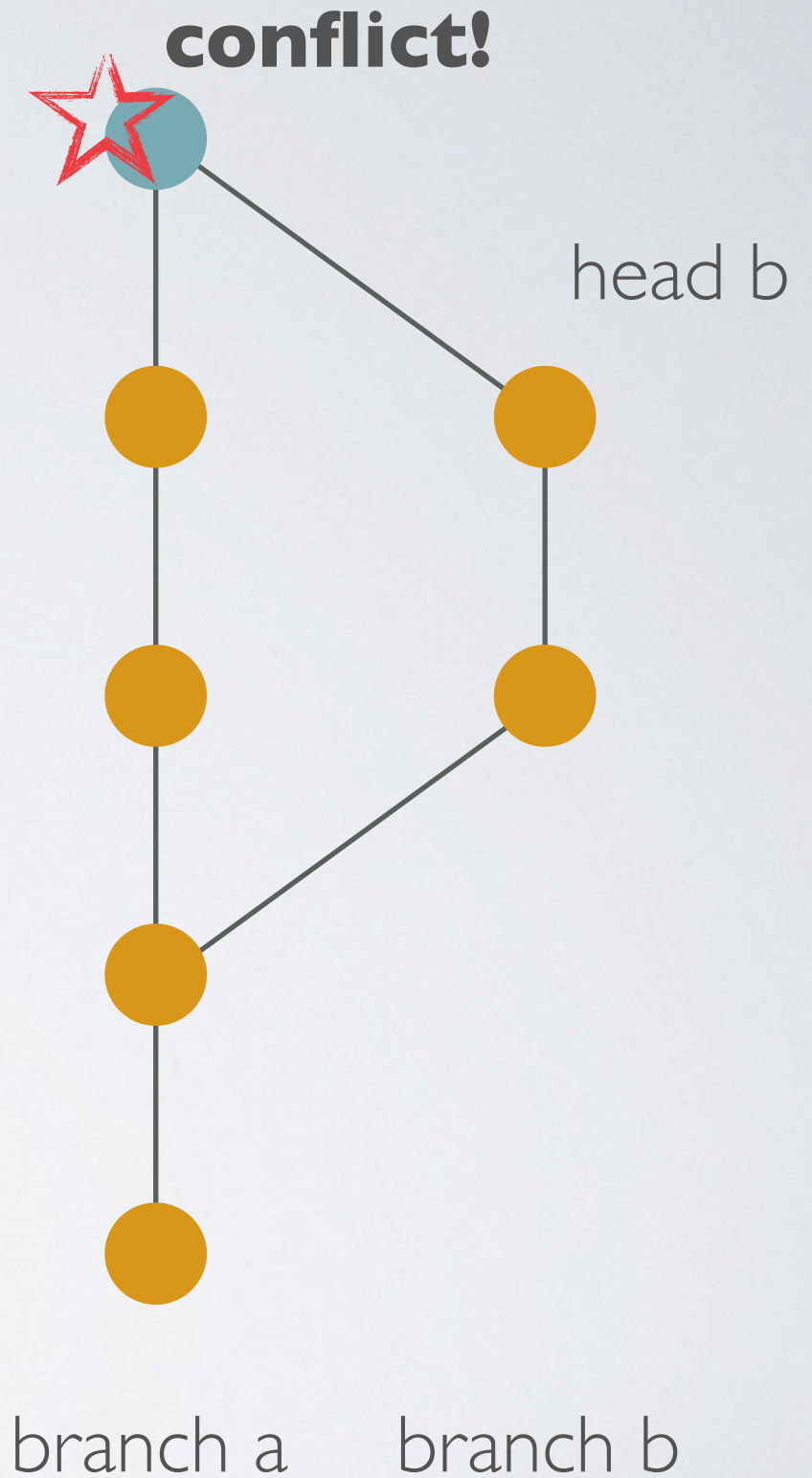
Conflicts



Conflicts

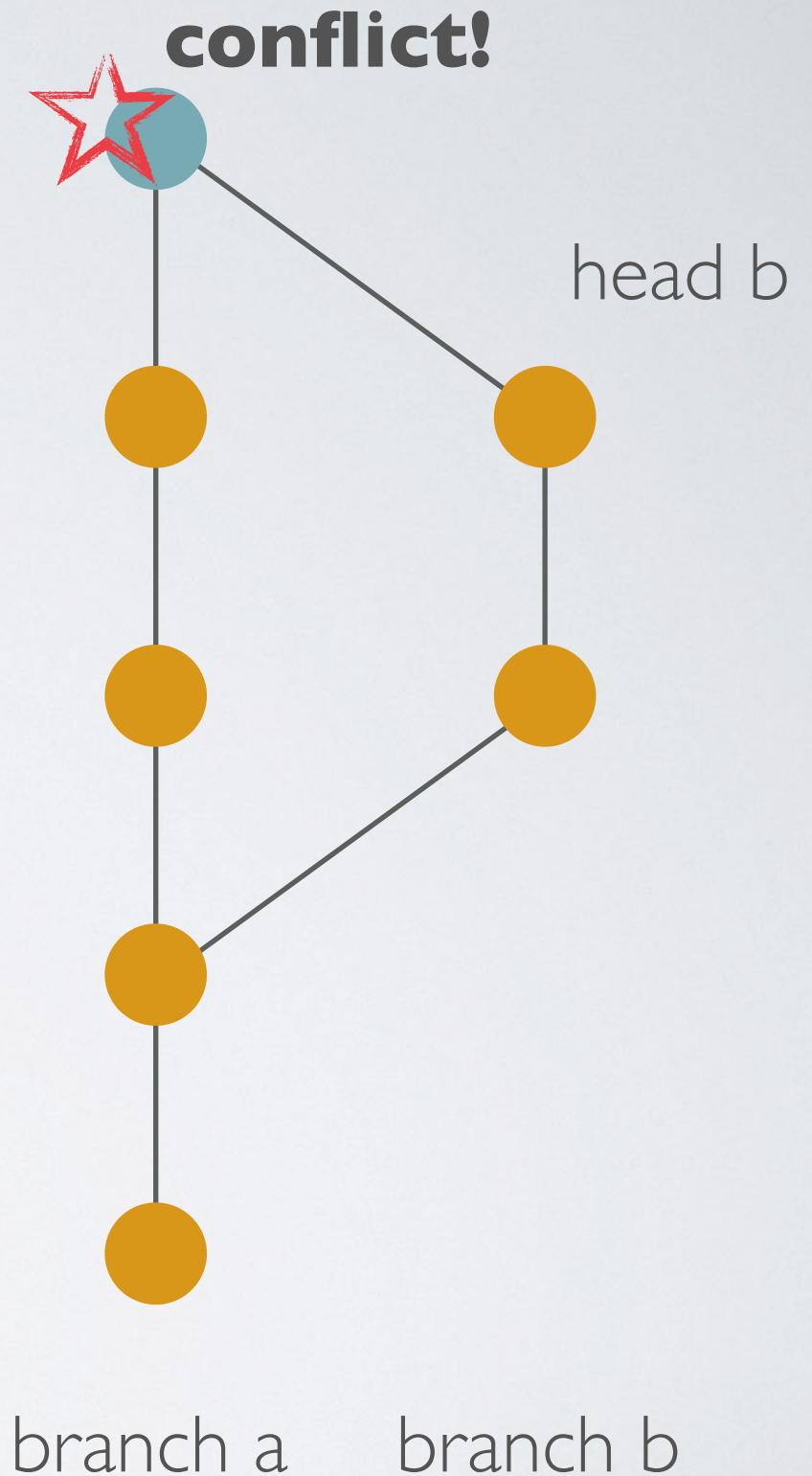


merge b into a



Conflicts

```
<<<<<<< HEAD  
void foo() {  
=====  
int foo() {  
>>>>>>> b  
    ...  
}
```



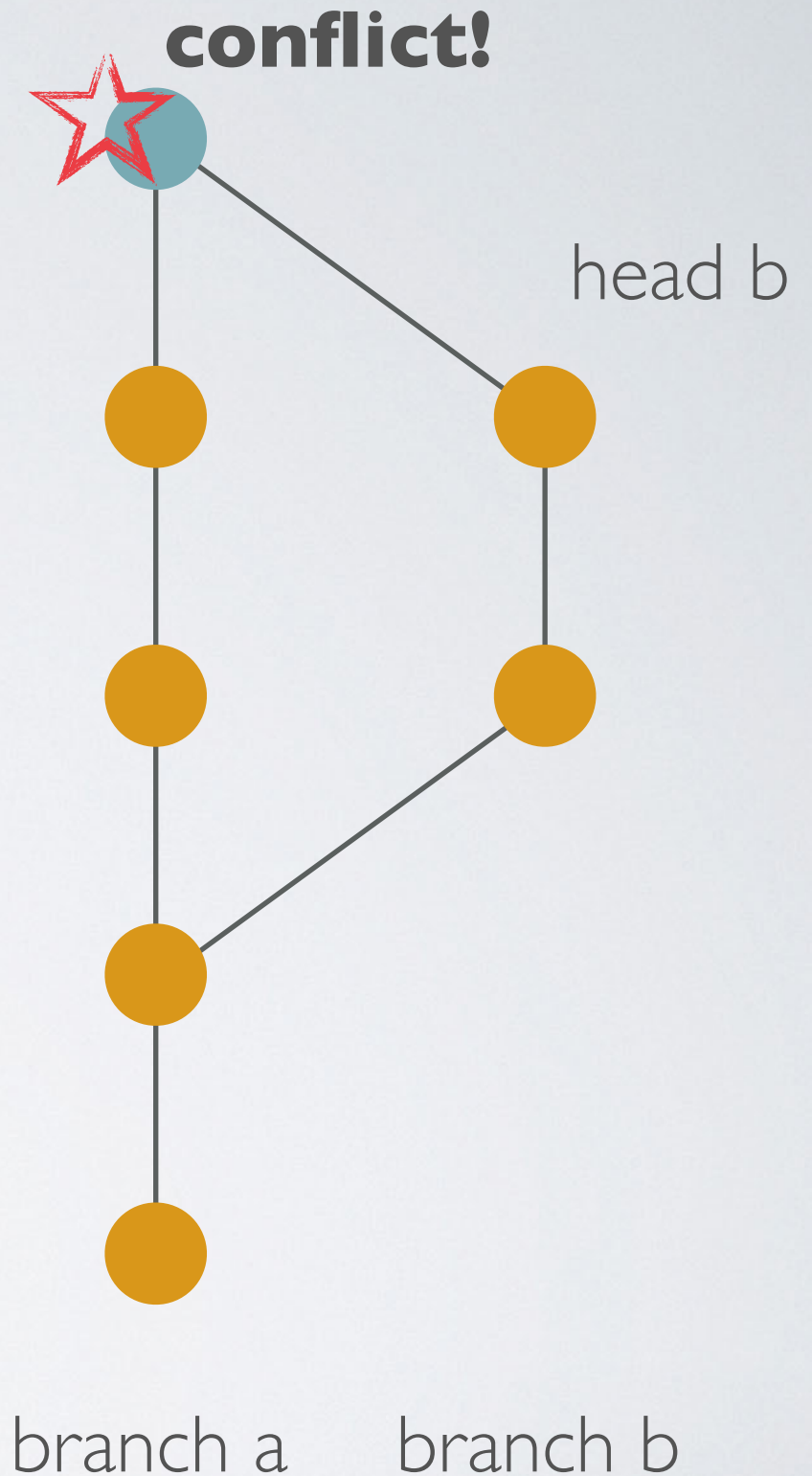
Conflicts

```
<<<<<<< HEAD
void foo() {
=====
int foo() {
>>>>>>> b
...
}
```

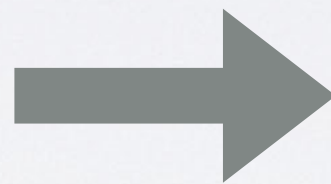
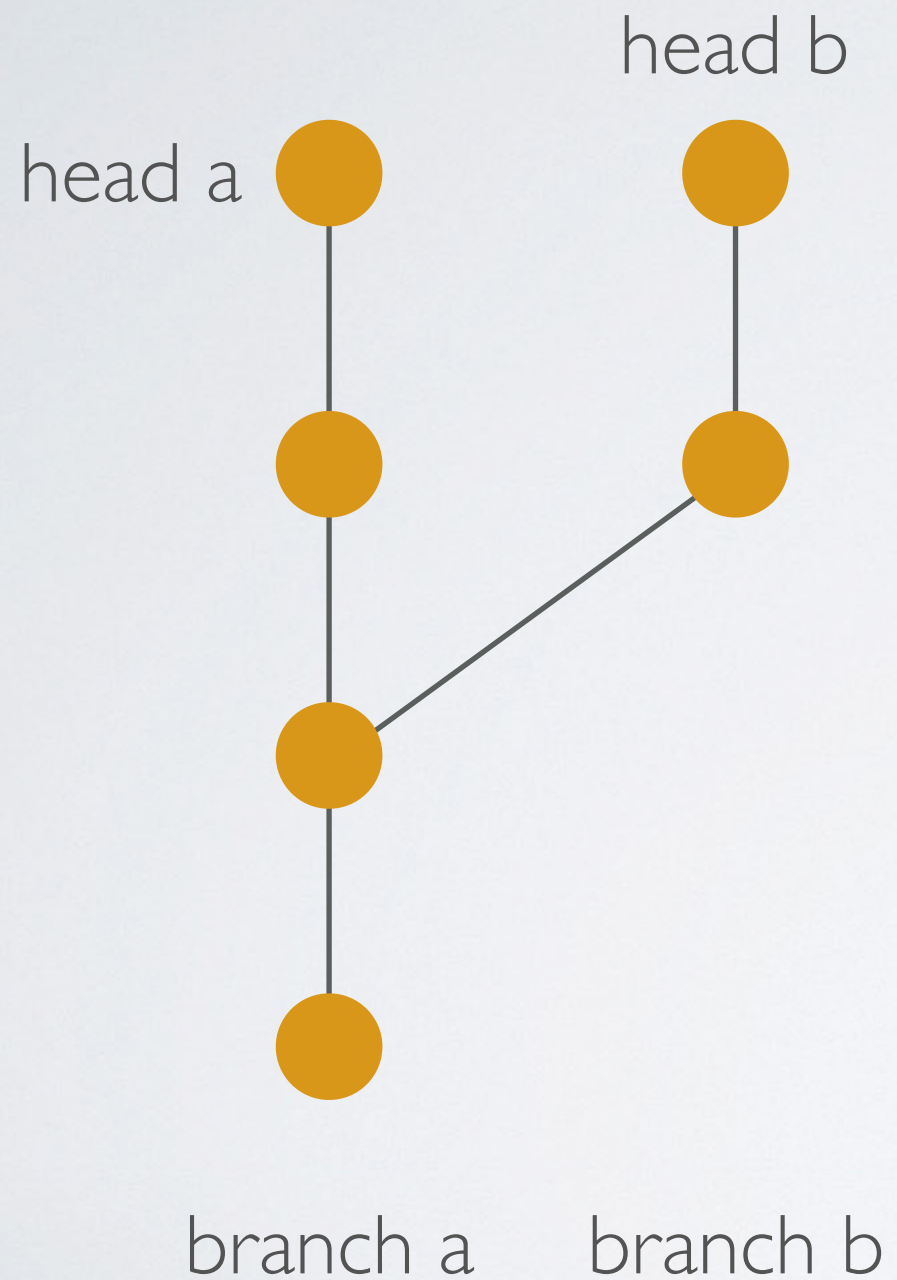


```
int foo() {
...
}
```

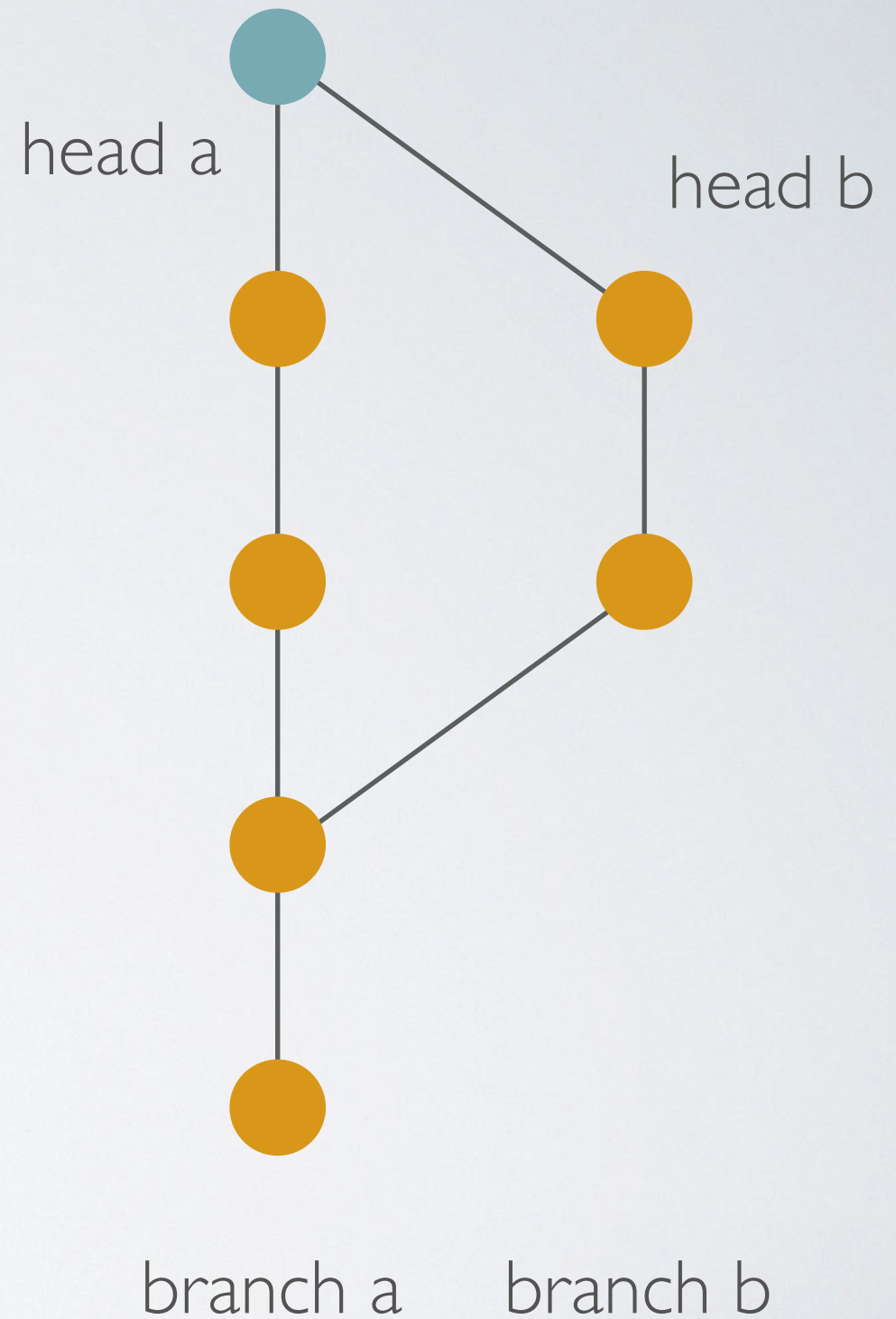
git add; git commit



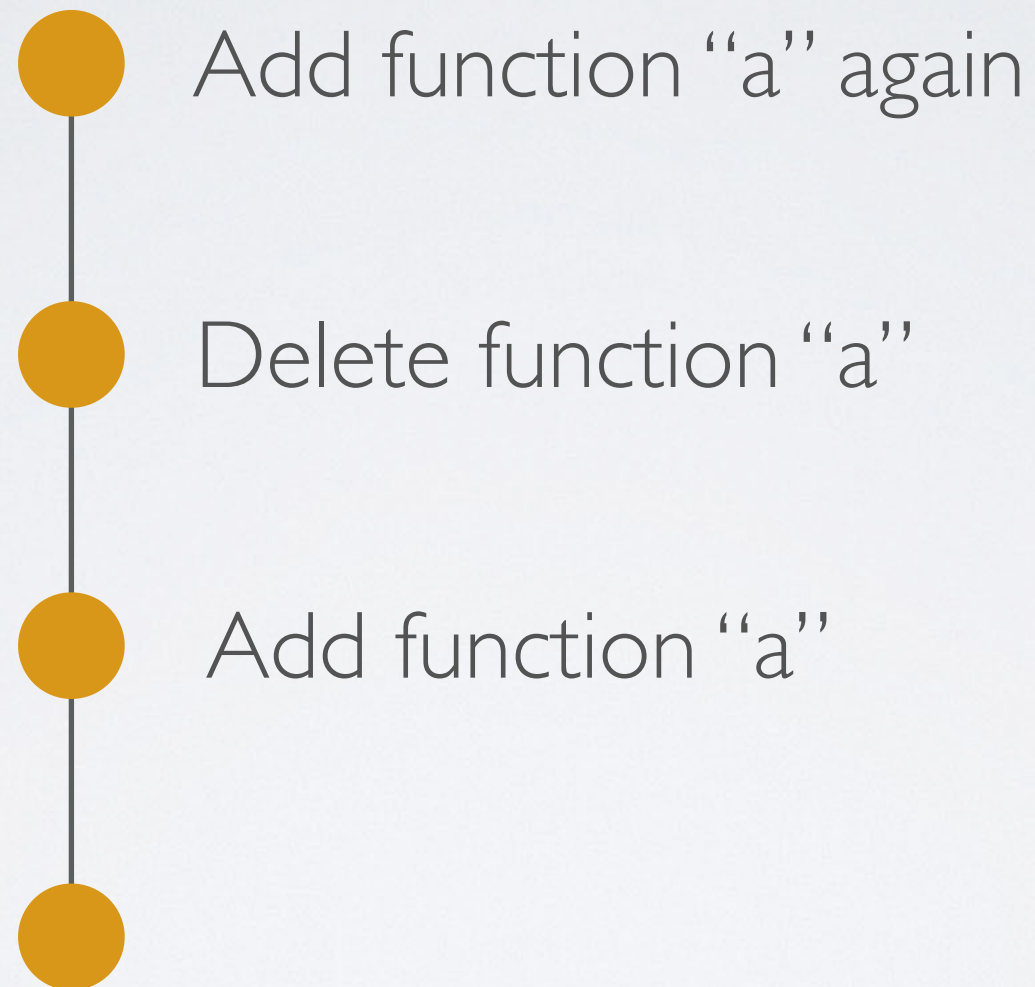
Conflicts



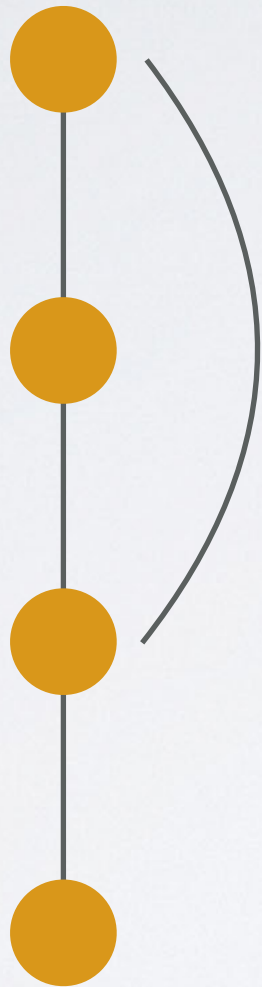
merge b into a



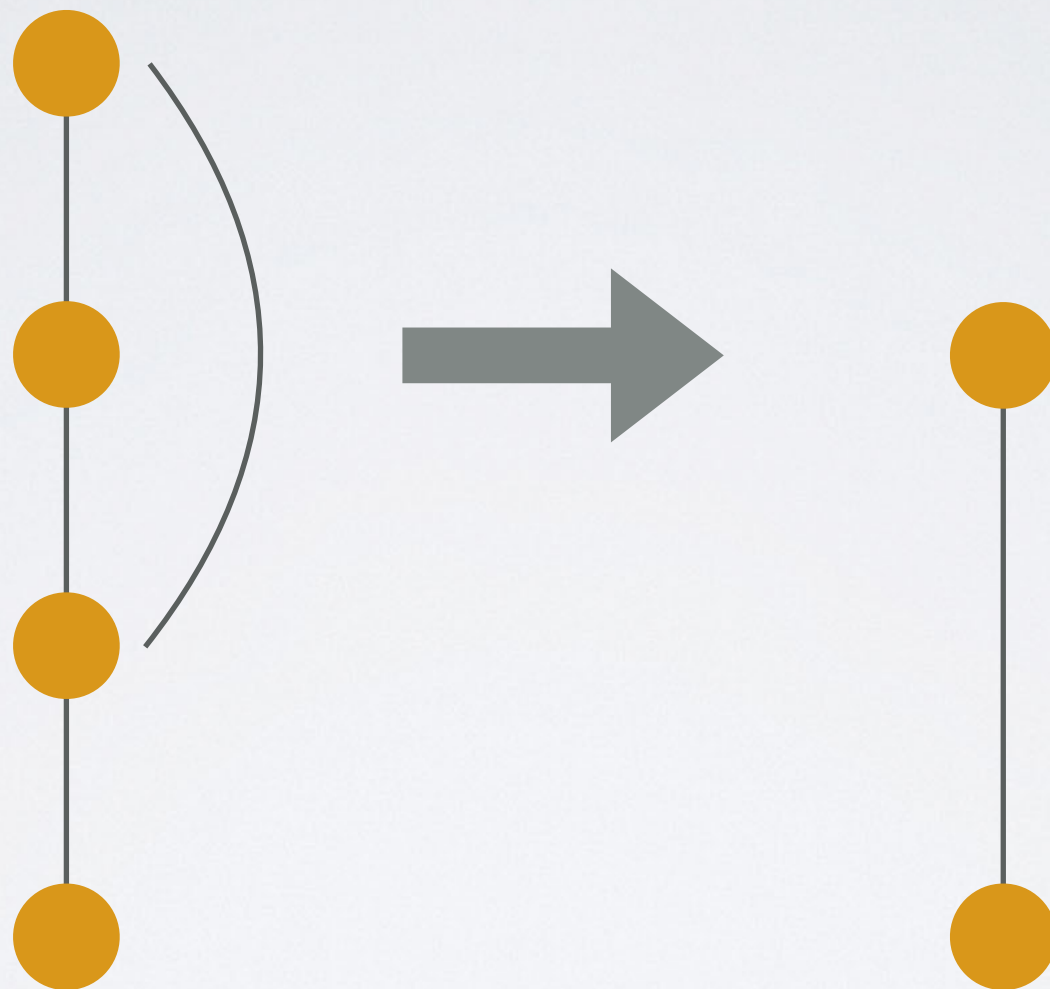
Squash



Squash



Squash

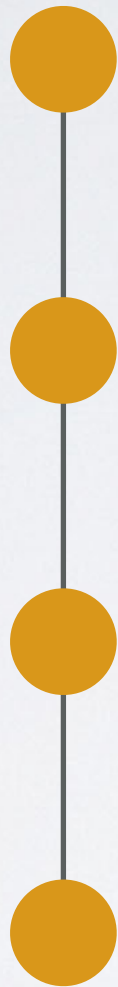


Squash

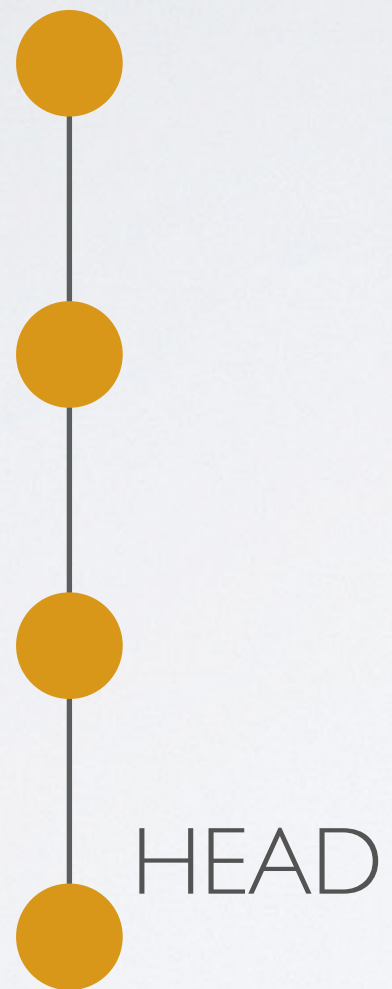
- `git reset <commit>`
 - reset head to specified commit
 - soft reset preserves the current files
- `git rebase -i <commit>`
 - modify existing commits

Squash

HEAD

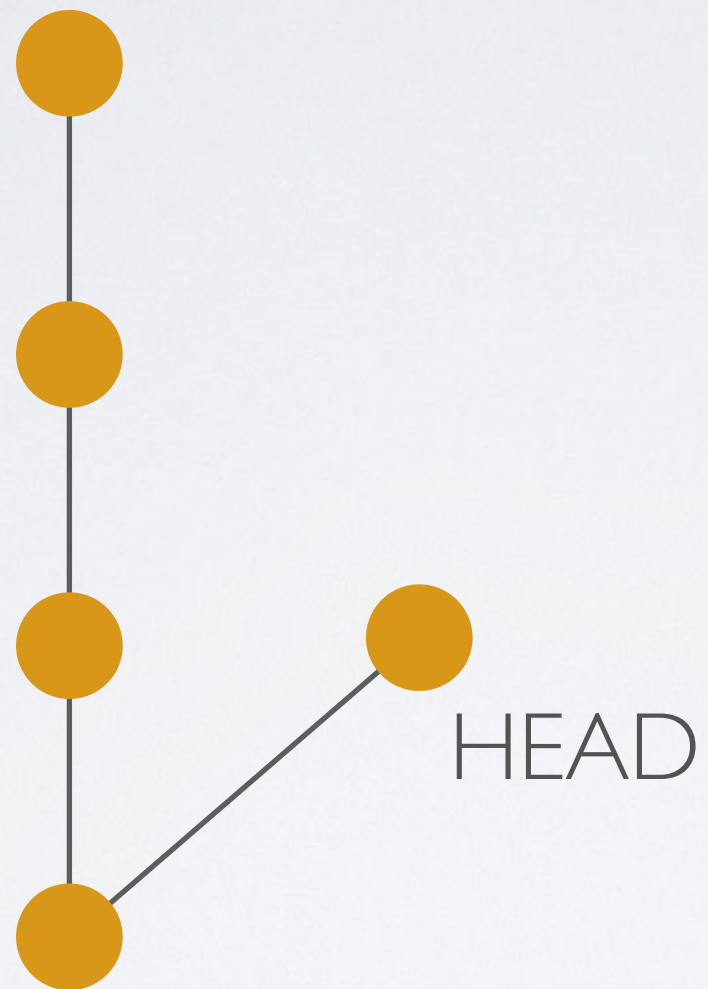


Squash



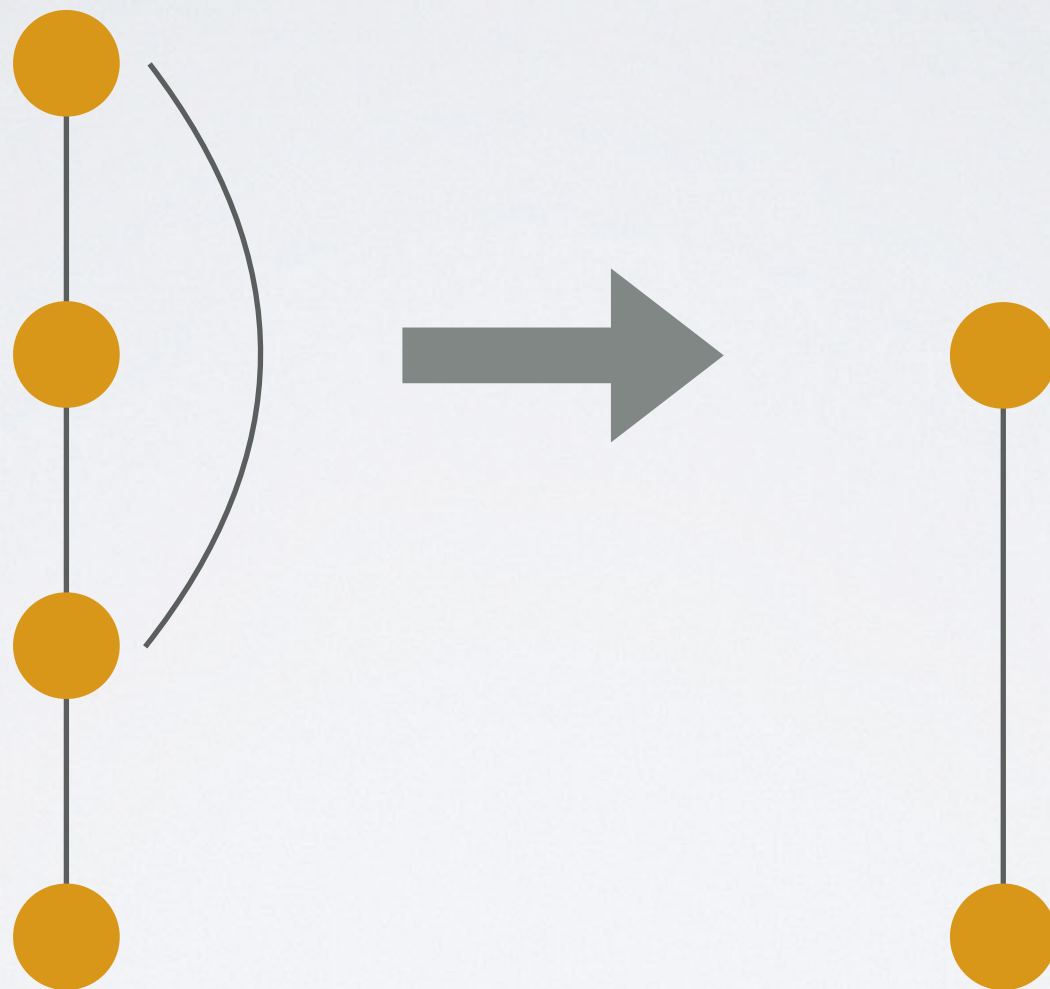
`git reset HEAD~3`

Squash



```
git add;  
git commit
```


Squash

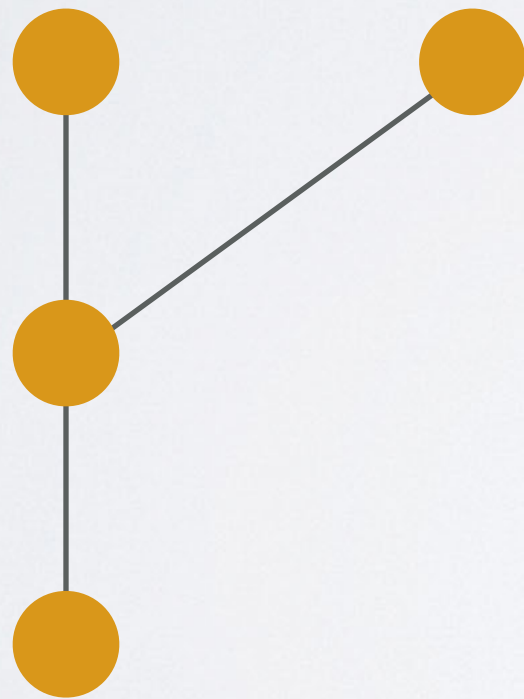


GitHub

- A git hosting repository (A web service)
 - remote repository
 - issues, pull requests

GitHub

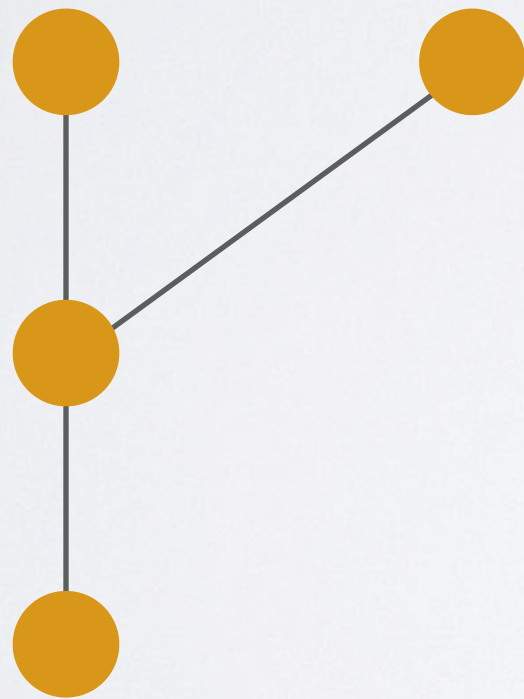
- A git hosting repository (A web service)
 - remote repository
 - issues, pull requests



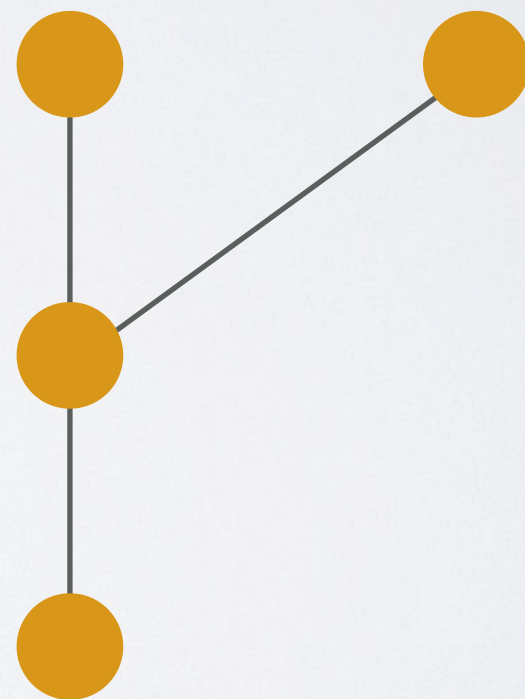
local

GitHub

- A git hosting repository (A web service)
 - remote repository
 - issues, pull requests



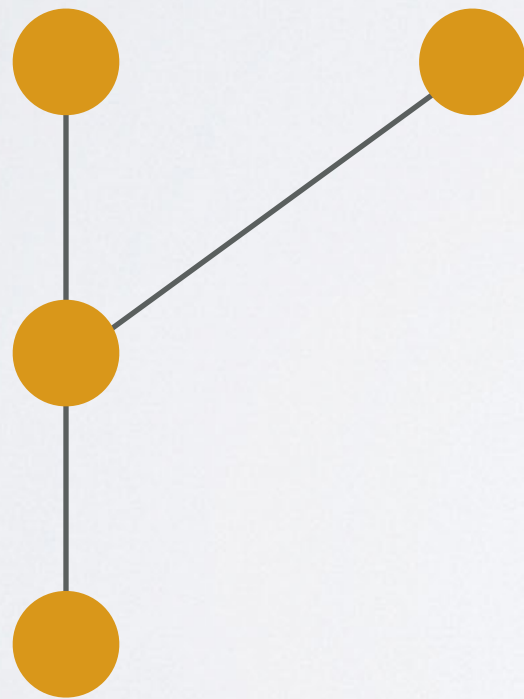
local



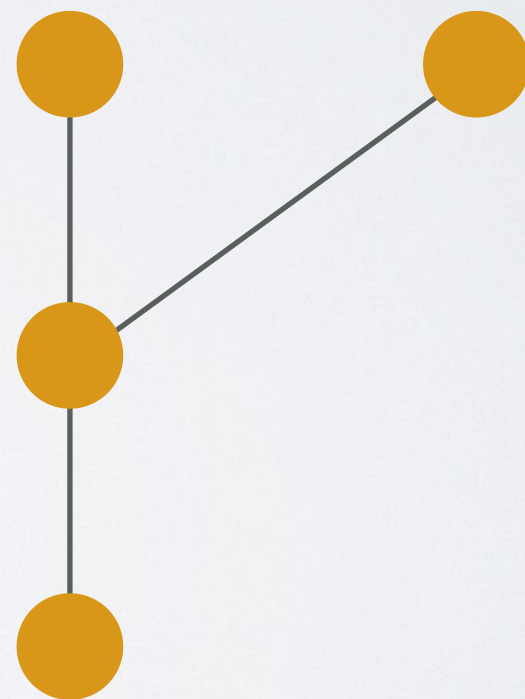
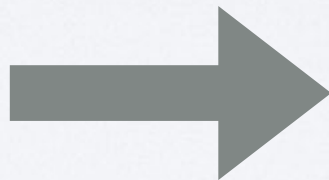
GitHub

GitHub

- A git hosting repository (A web service)
 - remote repository
 - issues, pull requests



local



GitHub

GitHub

- A git hosting repository (A web service)
 - remote repository
 - issues, pull requests
- **git clone** clones a remote repository
- **git push** updates the remote repository with the local commits
- **git fetch** fetches updates from the remote repository
- **git pull = git fetch + git merge**

Resources

- <https://try.github.io/>
- <https://guides.github.com/introduction/git-handbook/>
- <https://learngitbranching.js.org/>
- <https://git-scm.com/>
- <https://git-scm.com/book/en/v2>

TODO

- <https://github.com/snu-sf-class/swpp201901>
- <https://goo.gl/2ywCe8>
- Sign up for Github and submit this form (~3/12)