

Git Crash Course

Fearlessly develop!

Paul Bitutsky
Jerry Chen

Agenda

- Setup pre-reqs
- Why learn Git?
- Essential Git
- GitHub

What is Git?

Is it just short for GitHub?

- Answer: No
- Git is a version control system. It's a tool used to manage code history. It's like a programming language... it has a syntax
- GitHub is a website for hosting Git repositories
- An alternative to GitHub is Atlassian's Bitbucket, which also hosts Git repos

Why learn Git?

Why use version control?

The "real" question

- Many companies (and some courses @ Berkeley!) use a VCS (Version Control System)
- Git is one possible (and popular) tool

Companies & Projects Using Git

Google

facebook

Microsoft

twitter

LinkedIn

NETFLIX



PostgreSQL



Source: git-scm.com

File Home Insert Page Layout References Mailings Re

Cut Copy Paste Format Painter Clipboard

Calibri (Body) 11 A⁺ A⁻ Aa Aa




B *I* U ~~abc~~ x₂ x² A ab A

Font

Document Recovery

Word has recovered the following files. Save the ones you wish to keep.

Available files

_..w_v2.docx [Autosav...] Version created from the last ... 14:25 11 мая 2016 г.	
	_ v_v2.docx [Original] Version created last time the ... 14:17 11 мая 2016 г.	
	-ready11.doc [Original] Version created last time the ... 12:26 11 мая 2016 г. Repaired	

Document Recovery

Word has recovered the following files. Save the ones you wish to keep.

Available files



....._..w_v2.docx [Autosav...]
Version created from the last ...
14:25 11 мая 2016 г.



_ _ v_v2.docx [Original]
Version created last time the ...
14:17 11 мая 2016 г.







-ready11.doc [Original]
Version created last time the ...
12:26 11 мая 2016 г.

Repaired

Manage versions

Drive keeps older versions of 'disc11_sol.pdf' for 30 days. [Learn more](#)

UPLOAD NEW VERSION

-  Current version disc11_sol.pdf
Nov 30, 12:20 PM JERRY CHEN
-  Version 4 disc11_sol.pdf
Nov 27, 7:25 PM JERRY CHEN
-  Version 3 disc11_sol.pdf
Nov 26, 9:57 PM JERRY CHEN
-  Version 2 disc11_sol.pdf
Nov 26, 9:41 PM JERRY CHEN

Version history

Only show named versions ☐

Thursday

▼ January 25, 9:34 PM
Current version
■ JERRY CHEN

















01 - Control and Environments - Google Docs
January 25, 9:34 PM
■ JERRY CHEN

January 25, 9:32 PM
■ JERRY CHEN



















CLOSE

Can use different image

Past_Versions		
Name	Date Modified	Size
 Jerry Chen Resume v2.pdf	Jan 4, 2015 at 23:59	55 KB
 Jerry Chen Resume v3	Jan 12, 2015 at 13:02	55 KB
 Jerry Chen Resume v4.pdf	Jan 26, 2015 at 21:35	55 KB
 Jerry-Chen-Resume-1_26_17.pdf	Aug 27, 2016 at 22:09	89 KB
 Jerry-Chen-Resume-R-v1.pdf	Jul 31, 2015 at 16:53	67 KB
 Jerry-Chen-Resume-R-v15.pdf	Oct 12, 2015 at 19:45	85 KB
 Jerry-Chen-Resume-v5.pdf	Jan 26, 2015 at 21:35	55 KB
 Jerry-Chen-Resume-v6.pdf	Mar 19, 2015 at 11:19	56 KB
 Jerry-Chen-Resume-v7.pdf	Apr 16, 2015 at 14:47	55 KB
 Jerry-Chen-Resume-v8.pdf	Apr 16, 2015 at 14:51	55 KB
 Jerry-Chen-Resume-v9.pdf	Jul 5, 2015 at 15:44	57 KB
 Jerry-Chen-Resume-v10.1.pdf	Jul 26, 2015 at 21:50	85 KB
 Jerry-Chen-Resume-v10.pdf	Jul 6, 2015 at 18:16	57 KB
 Jerry-Chen-Resume-v11.pdf	Jul 28, 2015 at 21:10	67 KB
 Jerry-Chen-Resume-v12.pdf	Jul 30, 2015 at 14:56	67 KB
 Jerry-Chen-Resume-v13.pdf	Sep 4, 2015 at 18:13	84 KB
19 items, 122.67 GB available		

Can use different image

Past_Versions			
Past_Versions			
Name		Date Modified	Size
 Jerry Chen Resum v2.pdf		Jan 4, 2015 at 23:59	55 KB
 Jerry Chen Resum v3		Jan 12, 2015 at 13:02	55 KB
 Jerry Chen Resum v4.pdf		Jan 26, 2015 at 21:35	55 KB
 Jerry-Chen-Resum e-1_26_17.pdf		Aug 27, 2016 at 22:09	89 KB
 Jerry-Chen-Resum e-R-v1.pdf		Jul 31, 2015 at 16:53	67 KB
 Jerry-Chen-Resum e-R-v15.pdf		Oct 12, 2015 at 19:45	85 KB
 Jerry-Chen-Resum e-v5.pdf		Jan 26, 2015 at 21:35	55 KB
 Jerry-Chen-Resum e-v6.pdf		Mar 19, 2015 at 11:19	56 KB
 Jerry-Chen-Resum e-v7.pdf		Apr 16, 2015 at 14:47	55 KB
 Jerry-Chen-Resum e-v8.pdf		Apr 16, 2015 at 14:51	55 KB
 Jerry-Chen-Resum e-v9.pdf		Jul 5, 2015 at 15:44	57 KB
 Jerry-Chen-Resum e-v10.1.pdf		Jul 26, 2015 at 21:50	85 KB
 Jerry-Chen-Resum e-v10.pdf		Jul 6, 2015 at 18:16	57 KB
 Jerry-Chen-Resum e-v11.pdf		Jul 28, 2015 at 21:10	67 KB
 Jerry-Chen-Resum e-v12.pdf		Jul 30, 2015 at 14:56	67 KB
 Jerry-Chen-Resum e-v13.pdf		Sep 4, 2015 at 18:13	84 KB
19 items, 122.67 GB available			

Existing "Version Control"

Many problems!

- Not enough granularity
- *Too much* granularity
- Useless version naming
- Managing many copies of a file

Git

A bright and beautiful future

- Create "checkpoints" at your **own pace**
- **Custom descriptions** for versions
- Only a single version of a file to **focus** on at a time

Downside: This does take some time to learn...

Git Essentials

Vocabulary

- **Repository (Repo)** - a directory of files plus edit history
- **Commit** - a snapshot of the repository state

Some Non-Git Commands

Basic file manipulation

- `cat` will output contents of a file
- `touch file.txt` creates a empty file
- `echo "TEXT" > file.txt` replaces file with "TEXT"
- `echo "TEXT" >> file.txt` appends "TEXT" to a file

Let's get started

Please get your computer ready

This part is best if you have a computer to follow along with

If you don't, no worries — you can view the slides & notes later

Creating a Local Git Repo

- Most of Git is local
- Remote work involves just a few new commands

Demo

pangram1.txt

<empty>

pangram2.txt

<empty>

Unmodified

Modified

Staged

pangram1.txt

<empty>

pangram2.txt

<empty>

```
echo "...dog.." > pangram1.txt
```

Unmodified

Modified

Staged

pangram1.txt

...dog...

pangram2.txt

<empty>

```
echo "...Trebek..." > pangram2.txt
```

	Unmodified	Modified	Staged
--	------------	----------	--------

pangram1.txt			
--------------	--	--	--

pangram1.txt		...dog...	
--------------	--	-----------	--

pangram2.txt			
--------------	--	--	--

pangram2.txt		...Trebek...	
--------------	--	--------------	--

```
git add pangram1.txt
```

Unmodified

Modified

Staged

pangram1.txt

...dog...

pangram2.txt

...Trebek...

git commit

Unmodified

Modified

Staged

pangram1.txt

...dog...

pangram2.txt

...Trebek...

Commit Messages

A brief note

- Good commit messages are hard to come by
- Bad commit messages are worse than useless

758575d Merge pull request #26 from Berkeley-CS162/blargh
606878f some persistence working again
ba6af4d (**origin/subdirs**) Revert to the "so far" commit
b887e6d lol
85320da tons of tests pass
06e8ee0 grow-tell-persistence seems to pass
f6aa390 so far
748f040 Merge branch 'subdirs' of <https://github.com/Berkeley-CS162/group11> into subdirs
f232868 made some progress on vine
b549c97 Various fixes (see full commit message for more detail)
1b4efdd blsggtefd
8710759 stupid piece of
971d0ac stuff
13c2813 Update helper fxn name
47e930e dir open
58468d2 Prevent bad access from crashing the program
402a5ff Deny writes to dirs
4ae06eb trying to open dirs
d0f3044 accessing files in subdirs
ae5cd2b prep for '.' and '..'

f232868 made some progress on vine
b549c97 Various fixes (see full commit message
1b4efdd blsggtefd
8710759 stupid piece of
971d0ac stuff
13c2813 Update helper fxn name
47e930e dir open

1. Short and descriptive
2. Use imperative mood for the subject line
3. Addl. details (if any) go in the body

Log

History in depth

- Let's formalize the concept of the log
- Commits remember their parent

Demo

ec6d394 Add Trebek pangram

You are here (HEAD)



Beginning of time

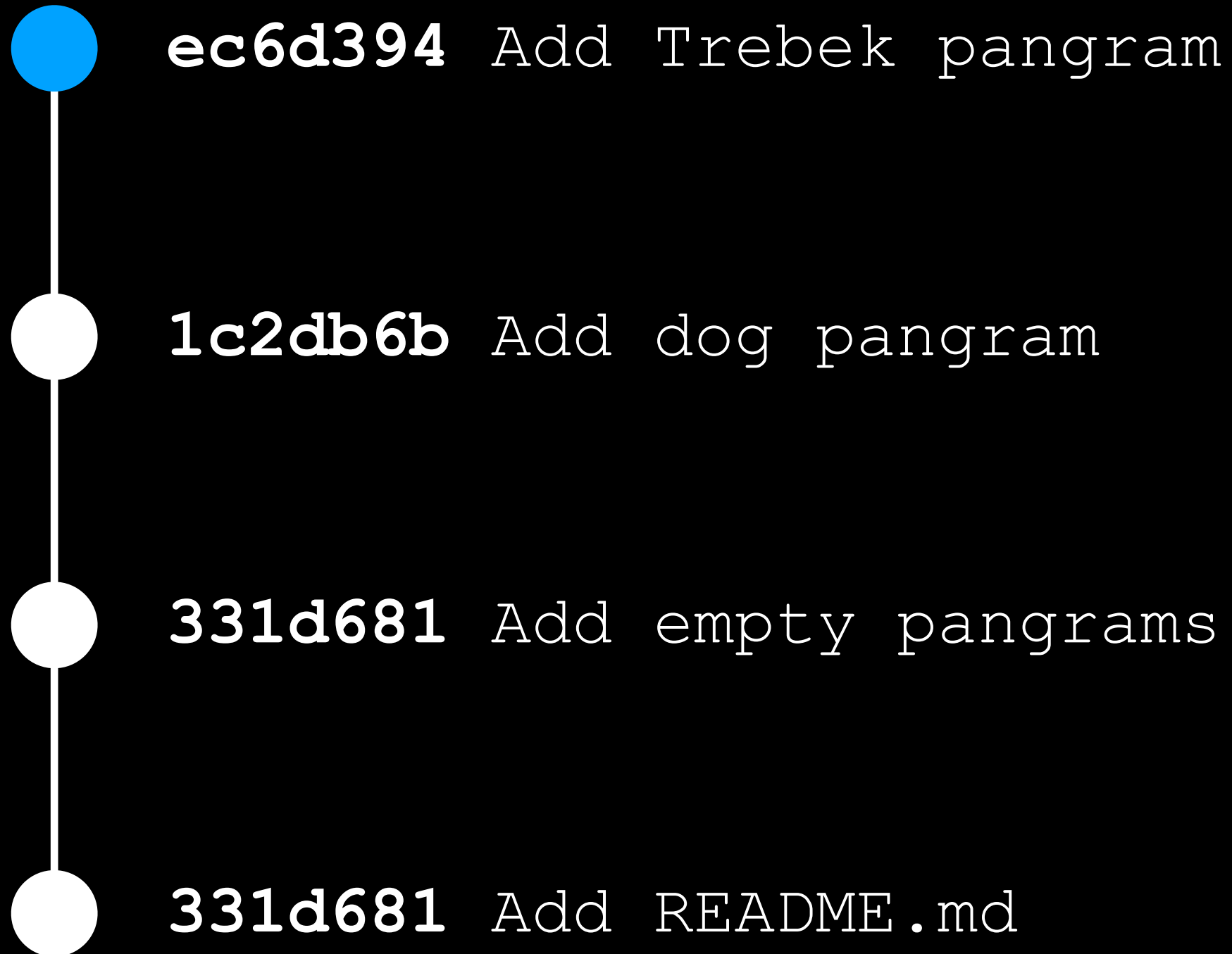
Branches

An introduction to multiverse theory

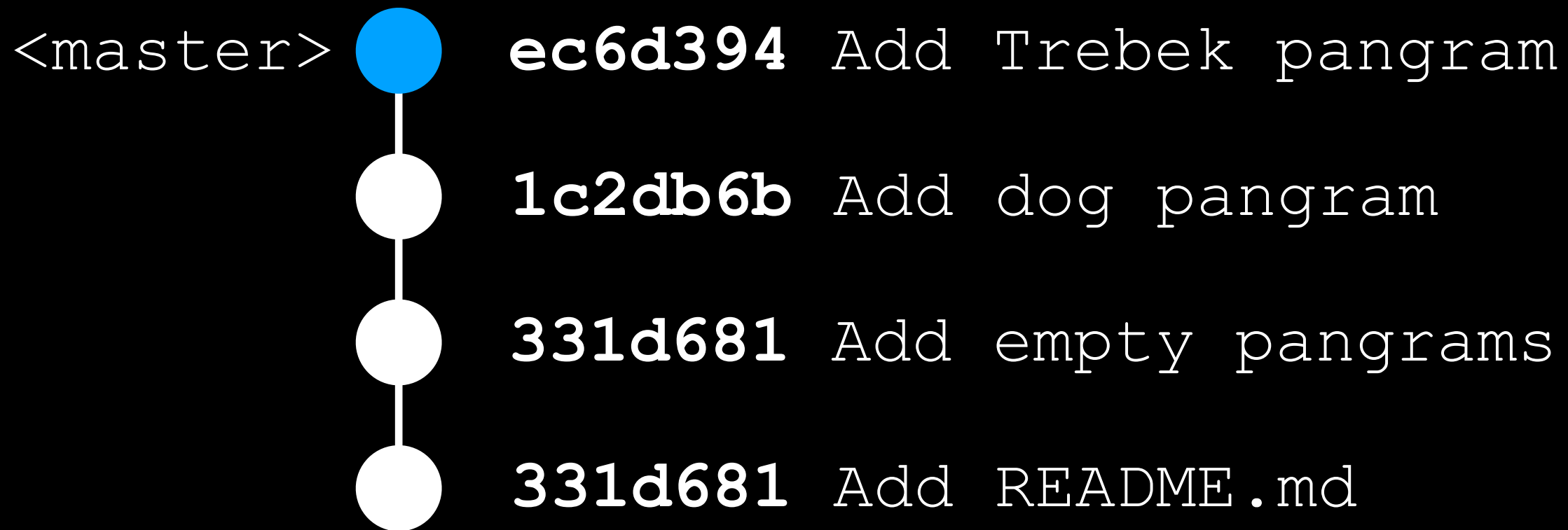
- Branches are labels pointing to specific commits
- "master" is the default starting branch

Demo

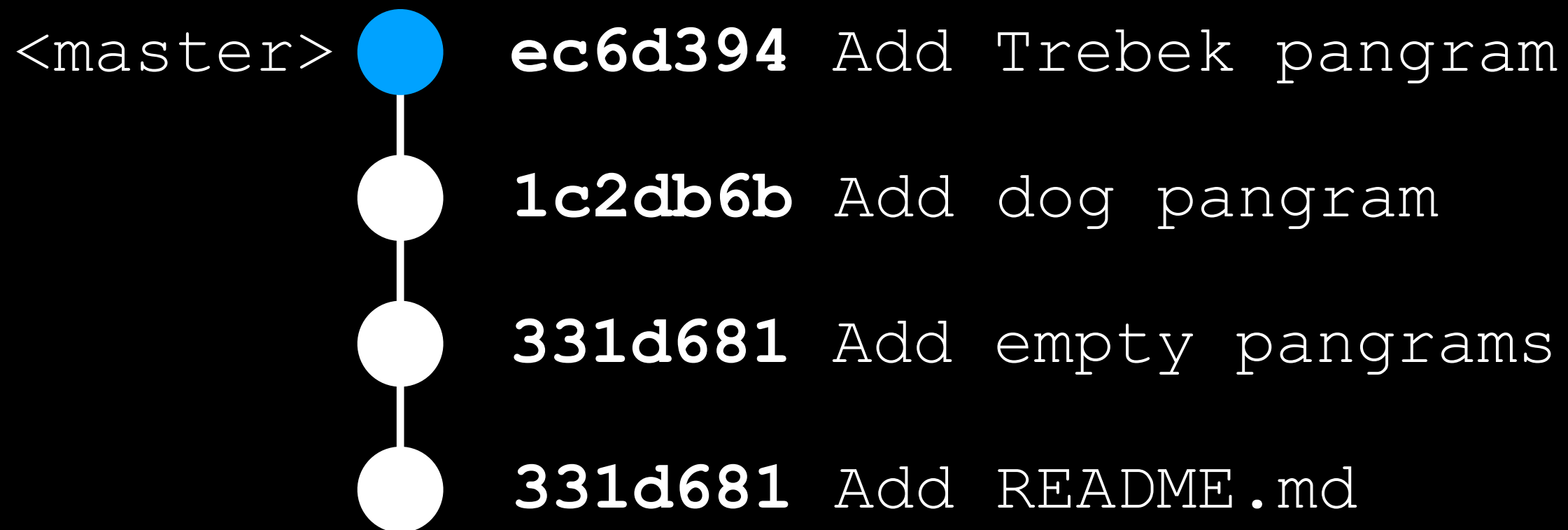
You are here



Beginning of time

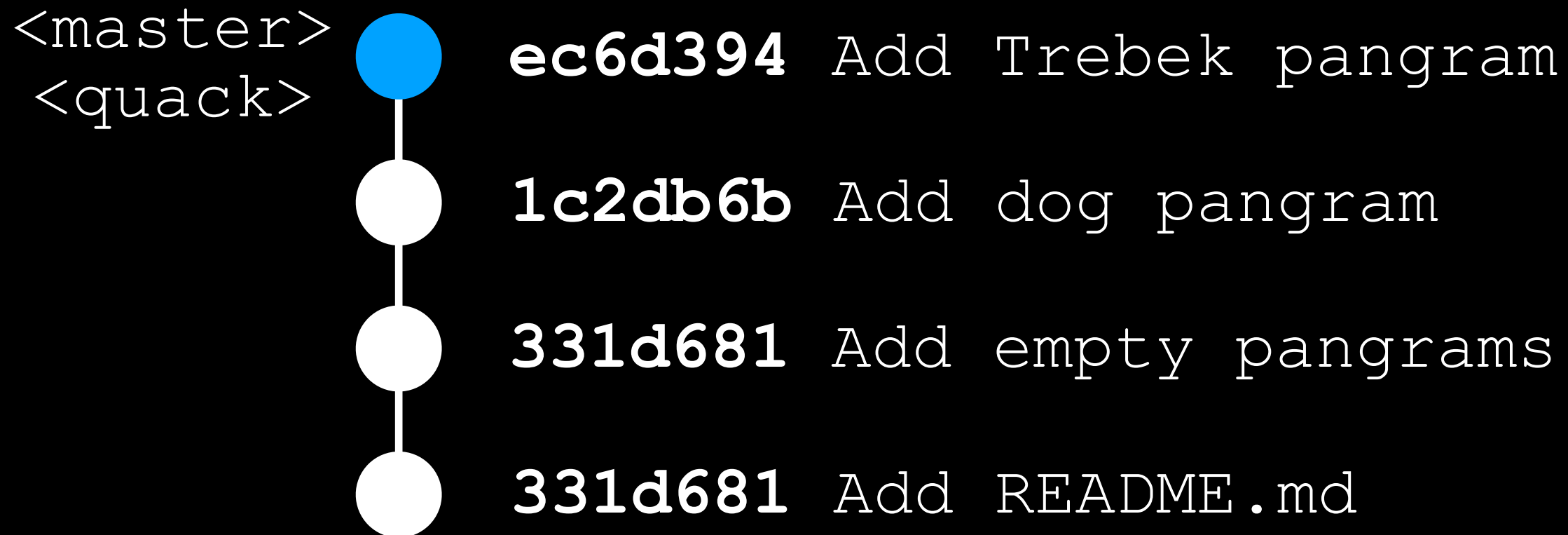


HEAD

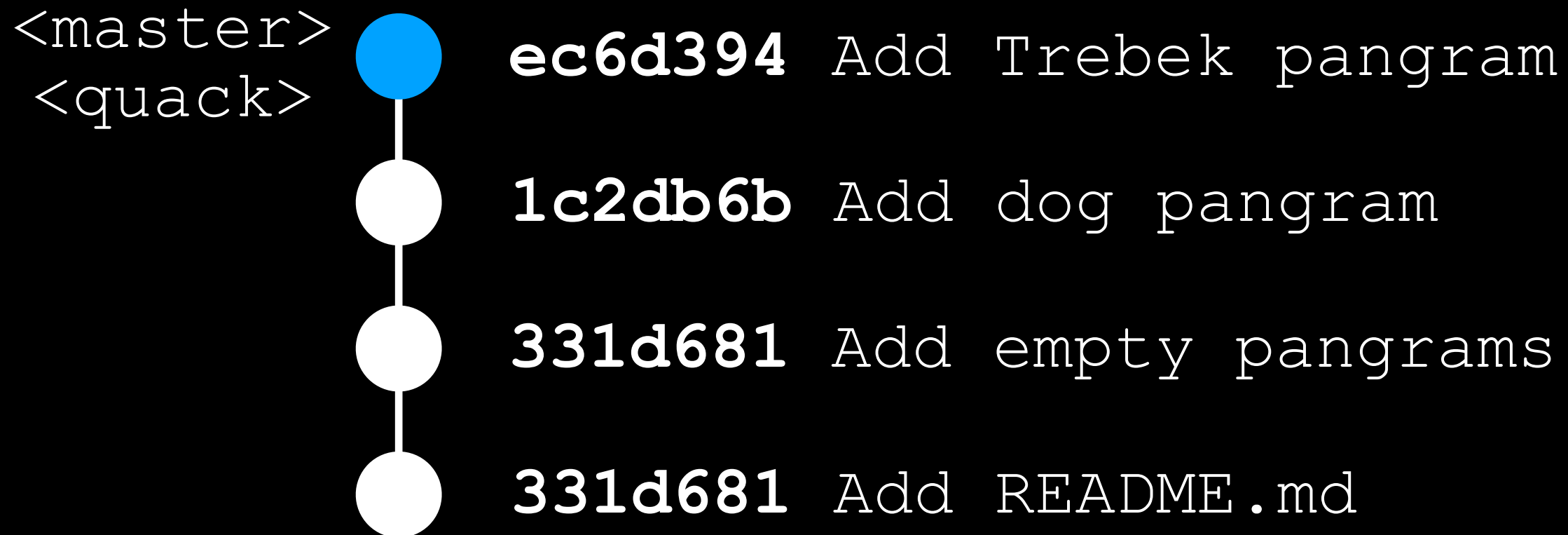


```
git branch quack
```

HEAD



git commit



git commit



```
git checkout quack
```



git commit



git commit



git commit



Merging

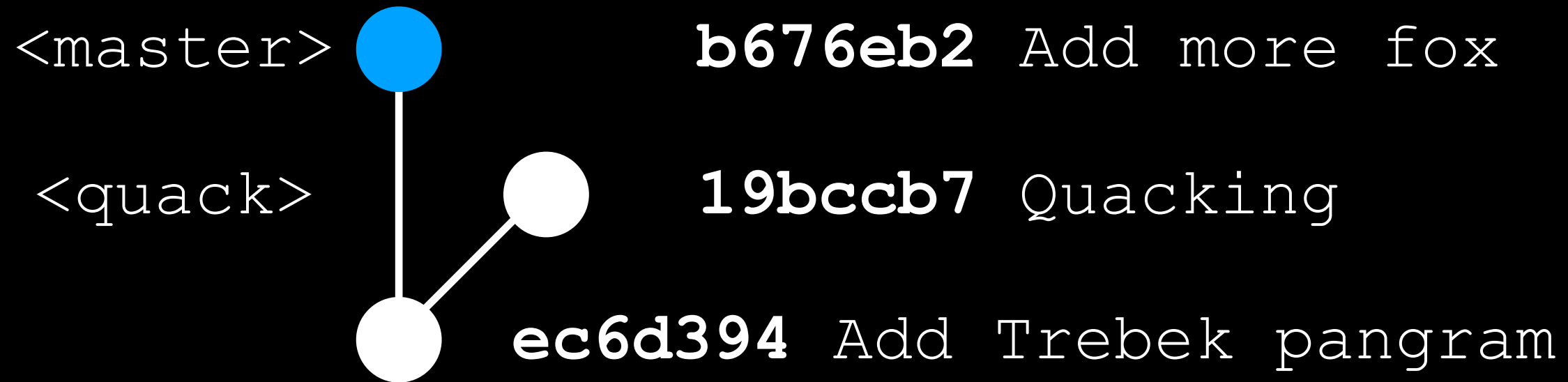
Worlds collide

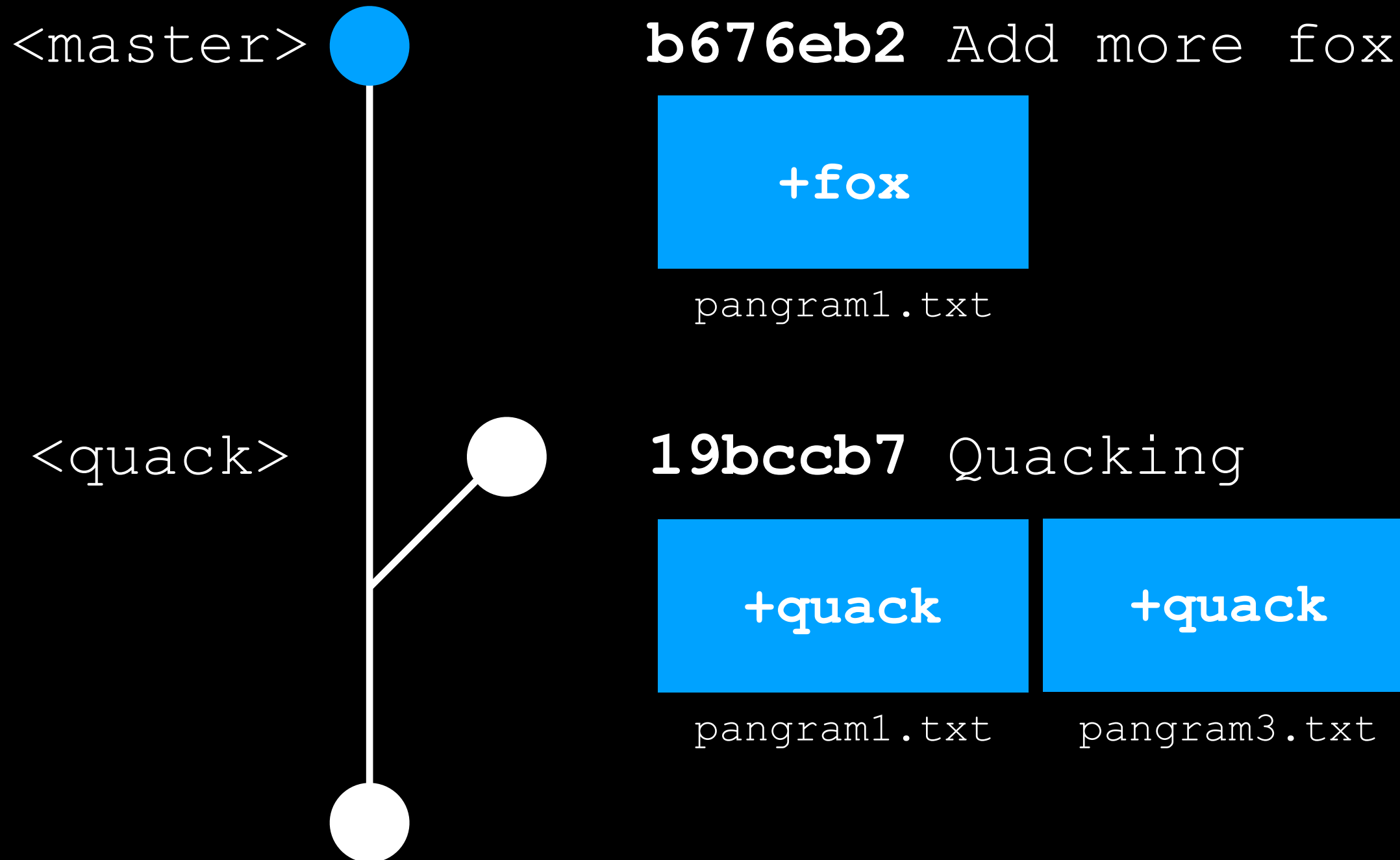
- Now it's time to combine our work
- Usually this is boring

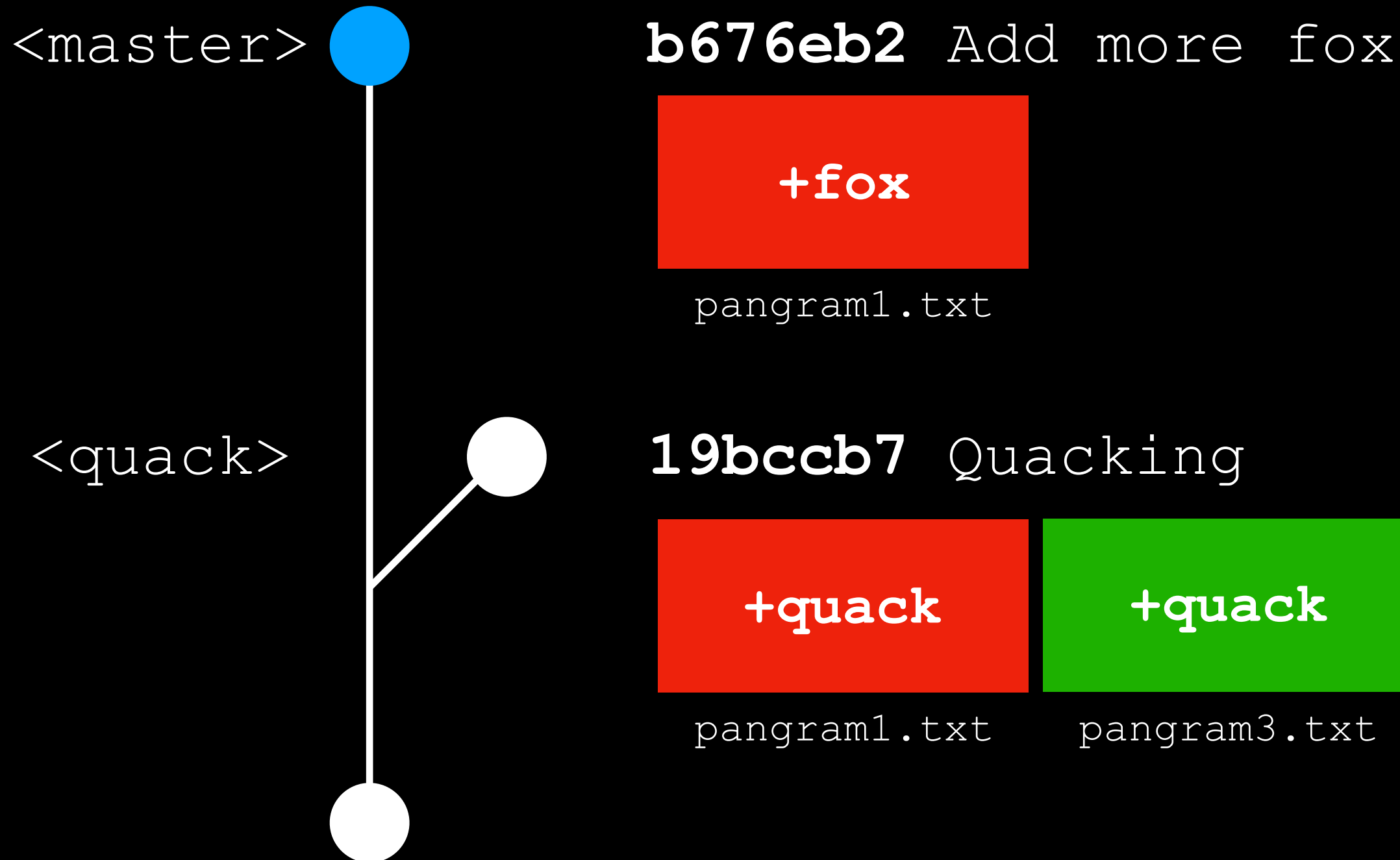
git checkout master



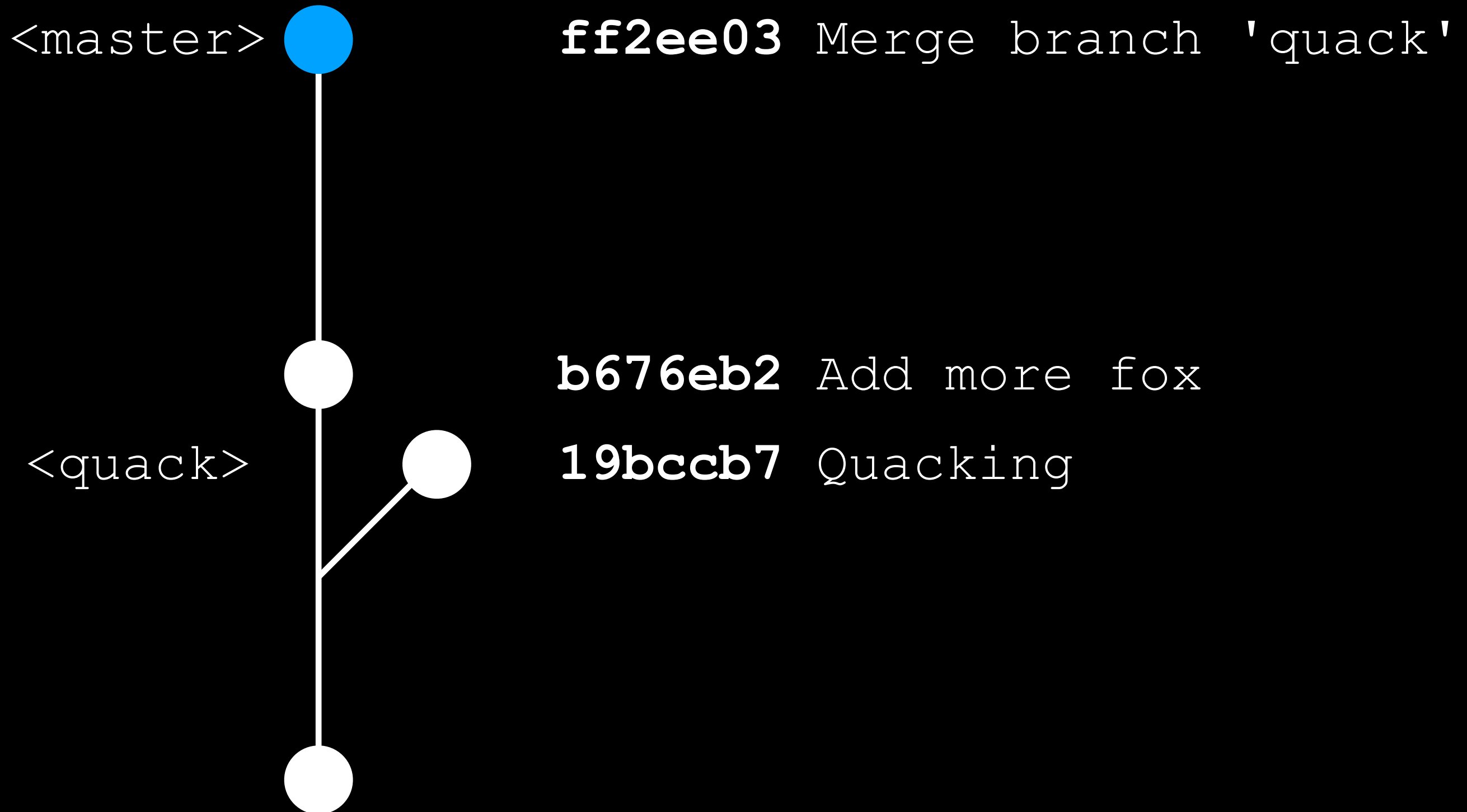
```
git checkout master
```







```
git merge quack
```



```
git merge quack
```

<master>

ff2ee03 Merge branch 'quack'

+fox
+quack

pangram1.txt

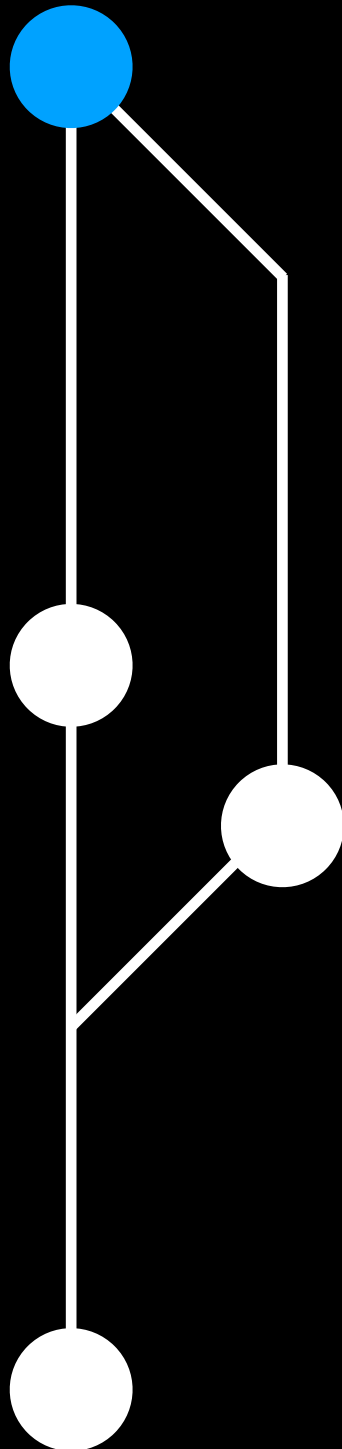
+quack

pangram3.txt

b676eb2 Add more fox

<quack>

19bccb7 Quacking



Demo

Break

Restoring

What good is a history if you can't use it?

- This part is dependent on the commits you created earlier
- As before, this works best if you can follow along

Checkout

One tool, many uses

Changing position in the commit history

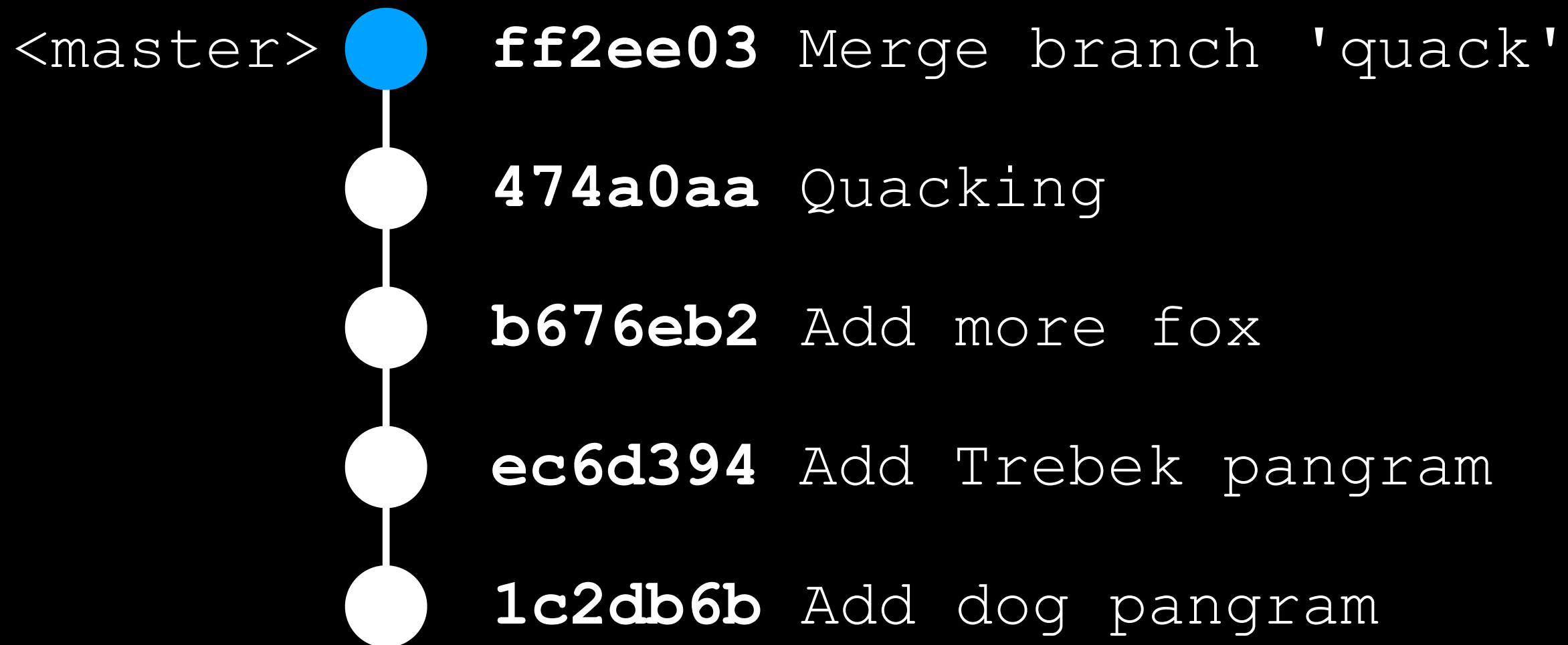
- `git checkout branch_name`
- `git checkout commit_id`

Changing files only

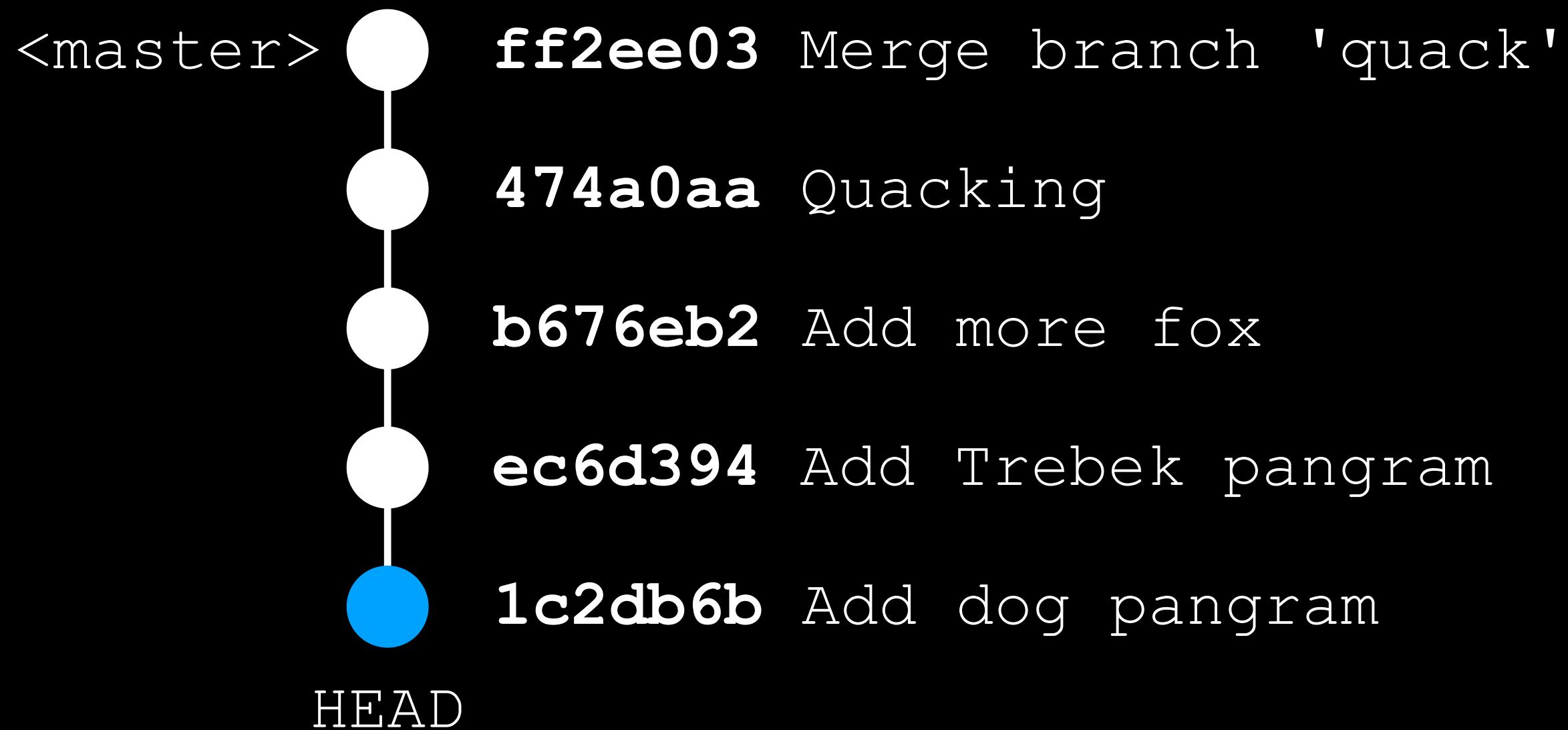
- `git checkout -- file_name`
- `git checkout commit_id -- file_name`

Demo

HEAD

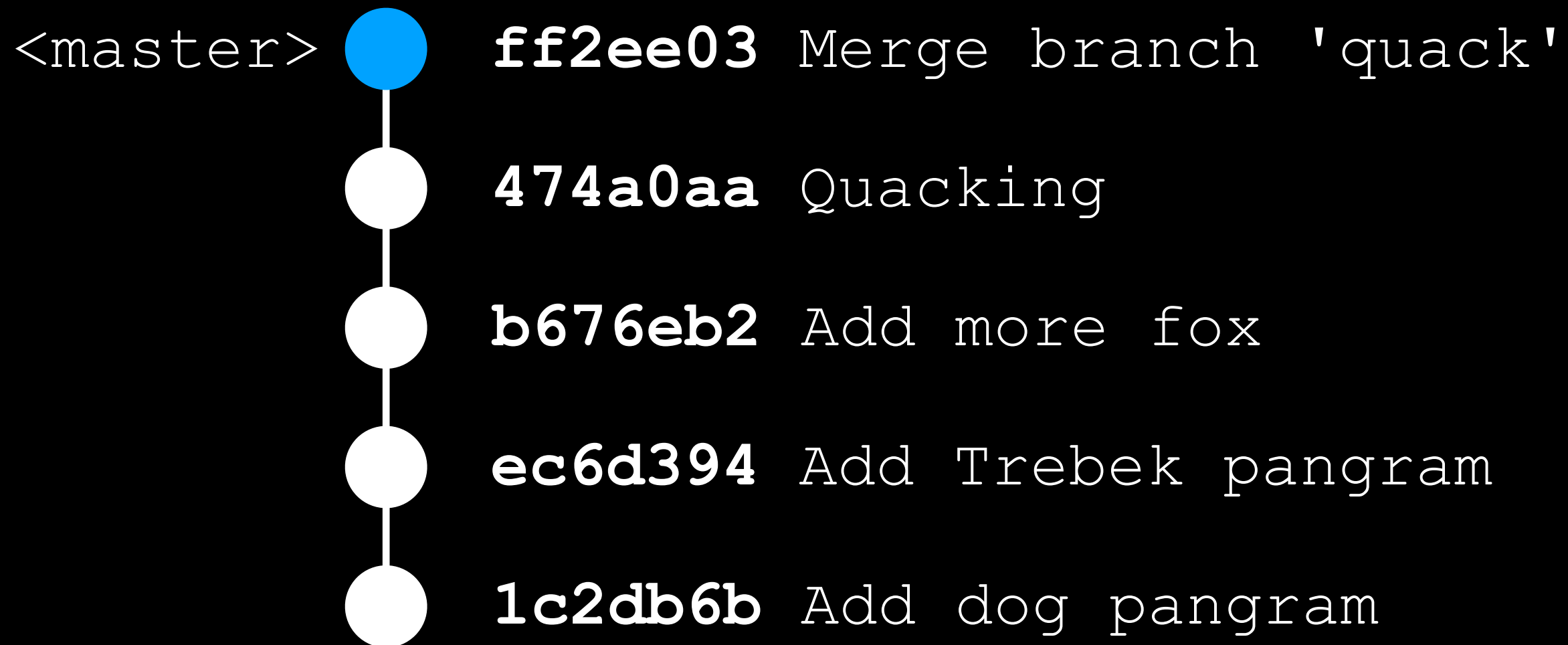


```
git checkout 1c2db6b
```



git checkout master

HEAD



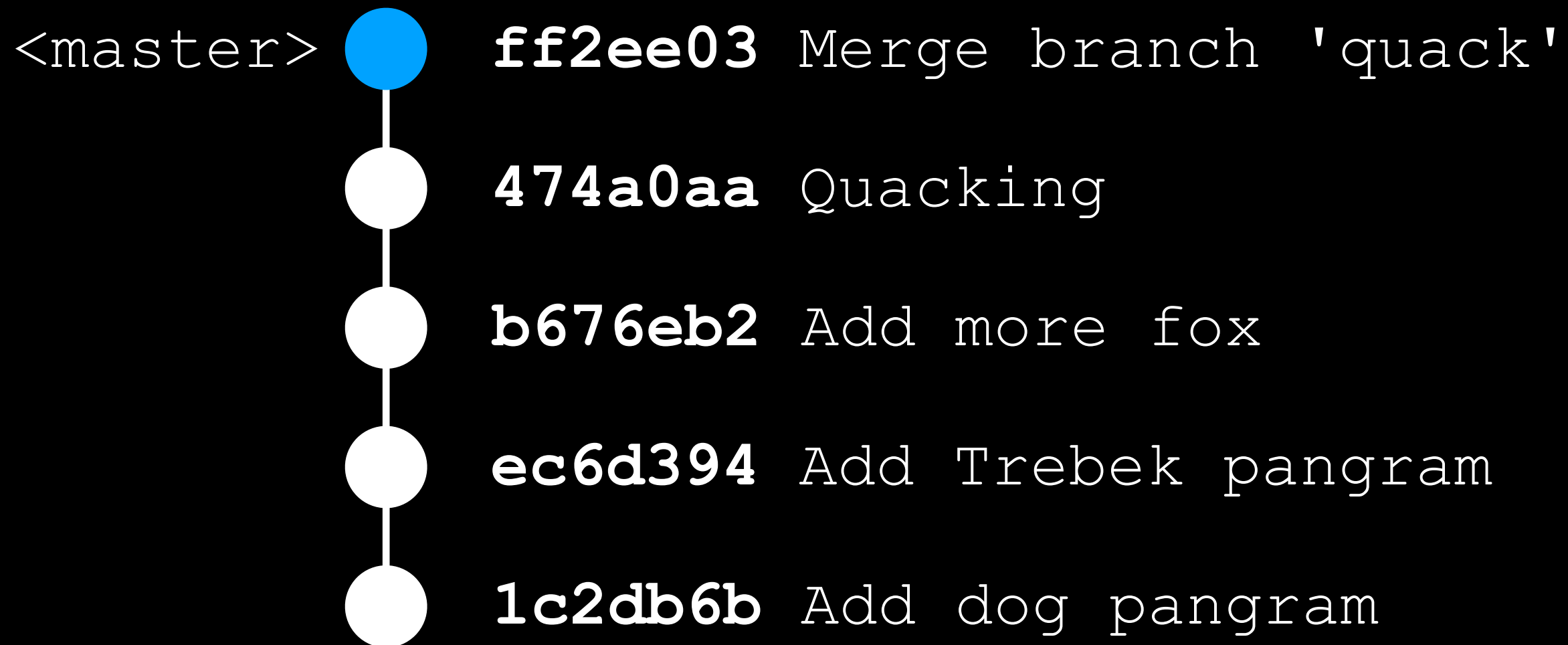
Reset

Change branches

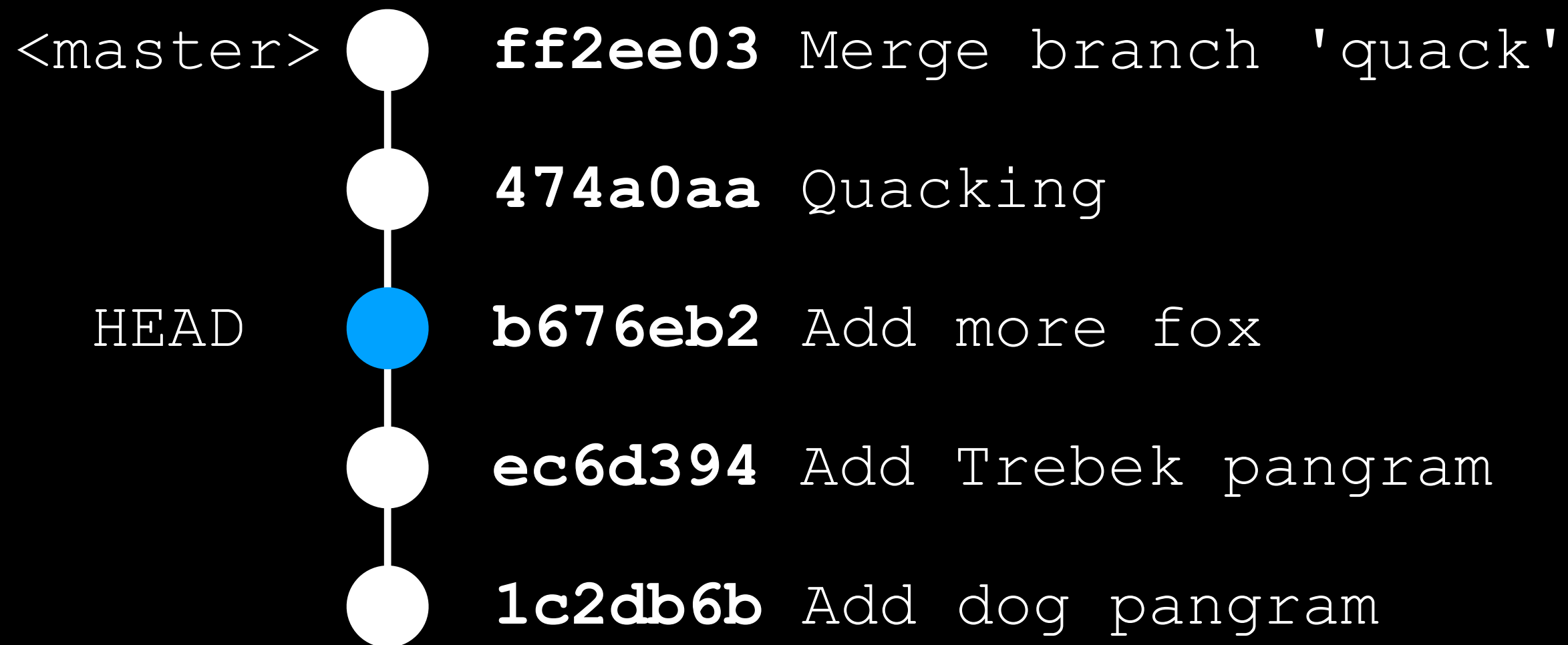
- Update the current head
- Also update the current branch to match (if applicable)

Demo

HEAD

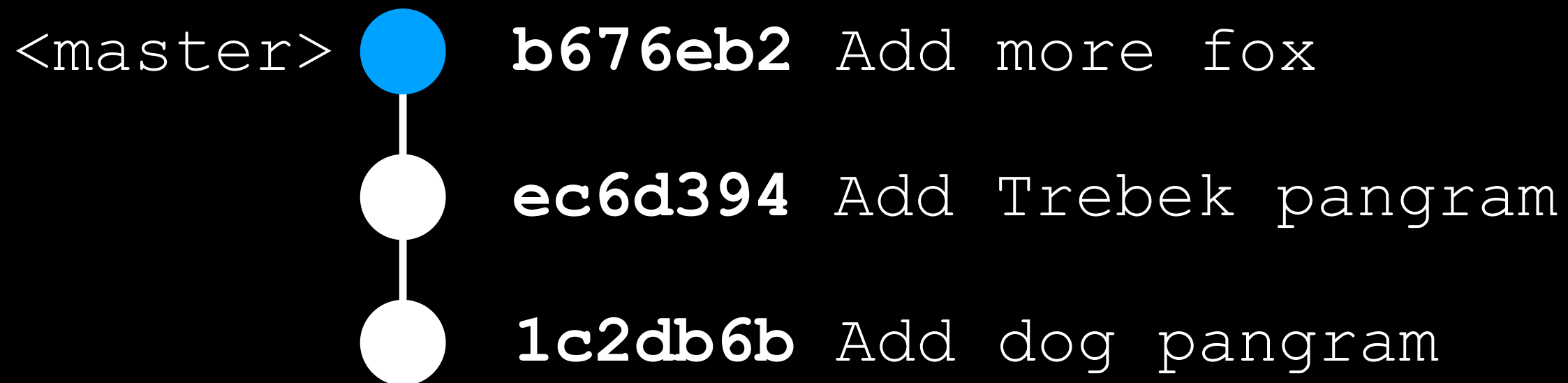


```
git reset b676eb2
```




```
git reset b676eb2
```

HEAD



Reset vs Checkout

When applied to branches (not files):

- checkout changes the location of HEAD
- reset changes the location of HEAD AND updates branch

Grab Bag

Three quick and useful tools

- `git diff`
- `git show`
- `git grep`

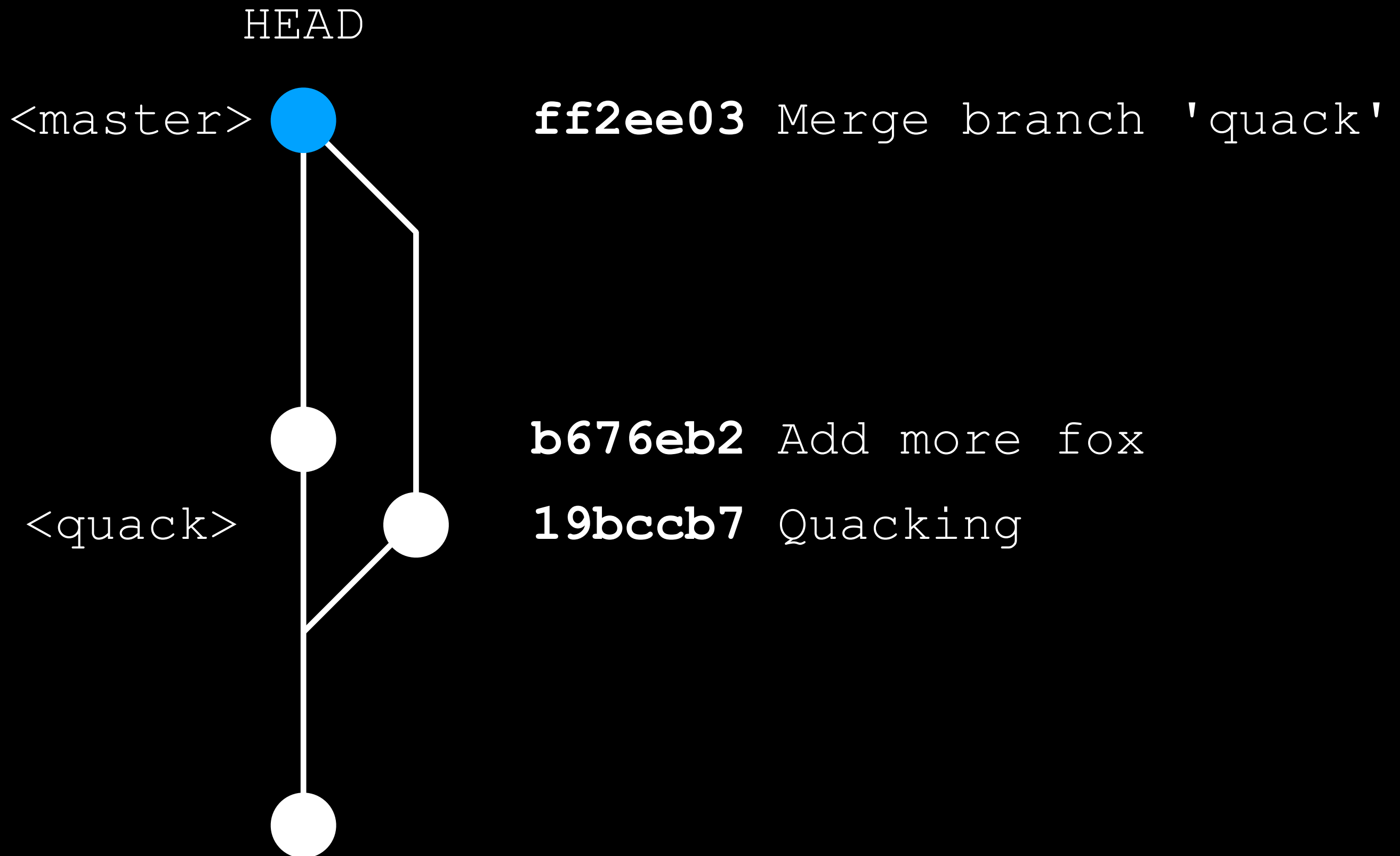
Demo

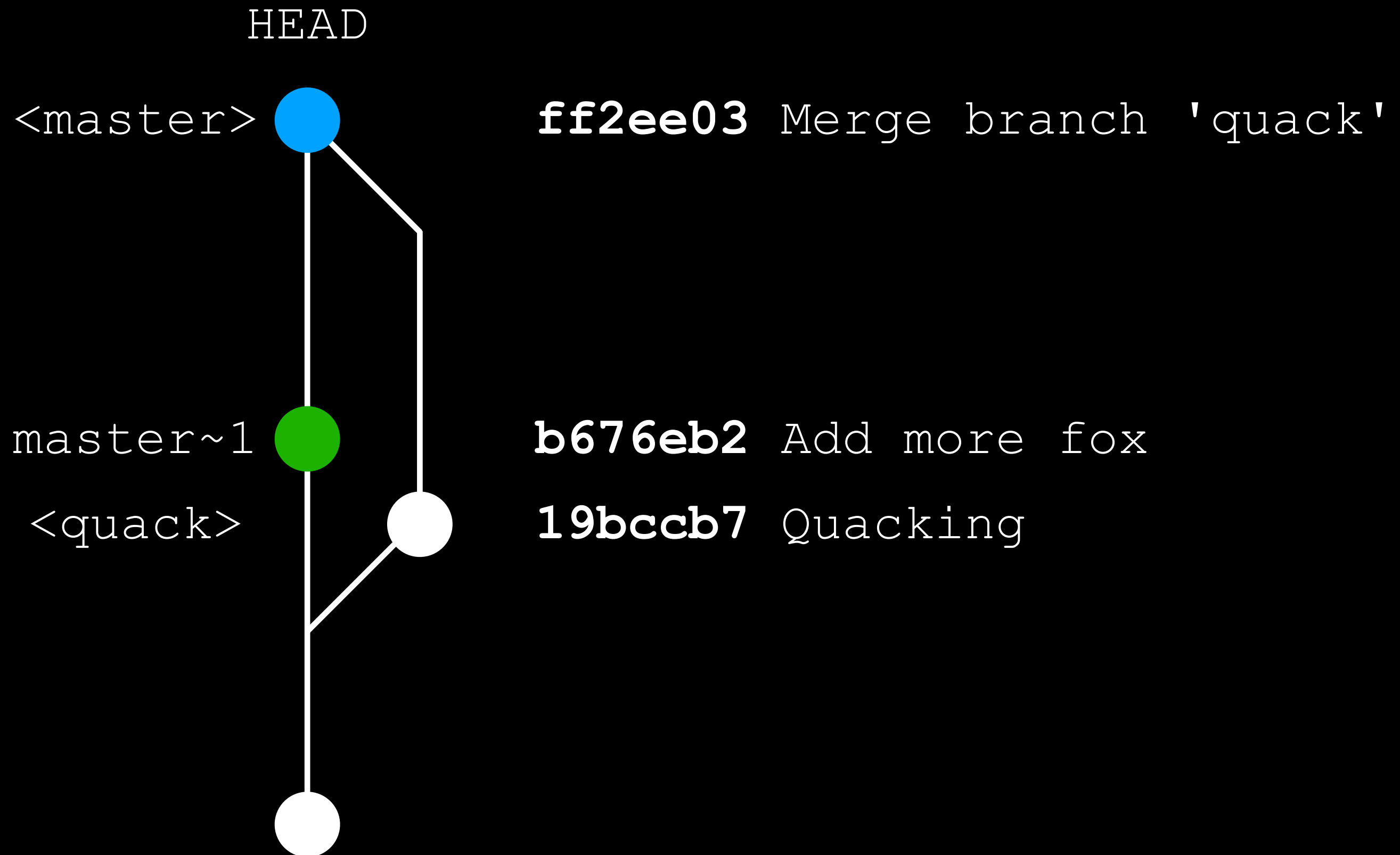
Grab Bag

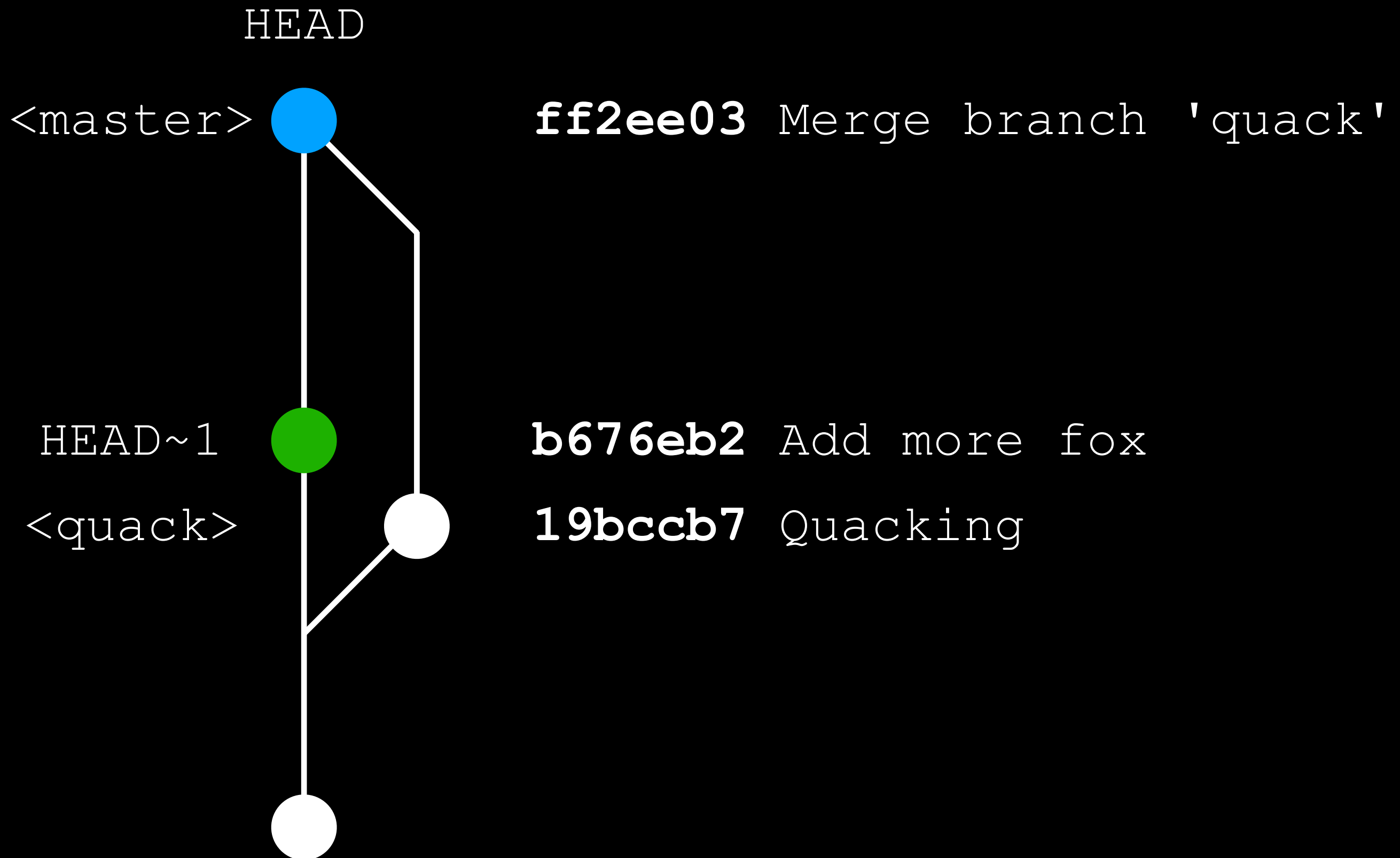
Relative references

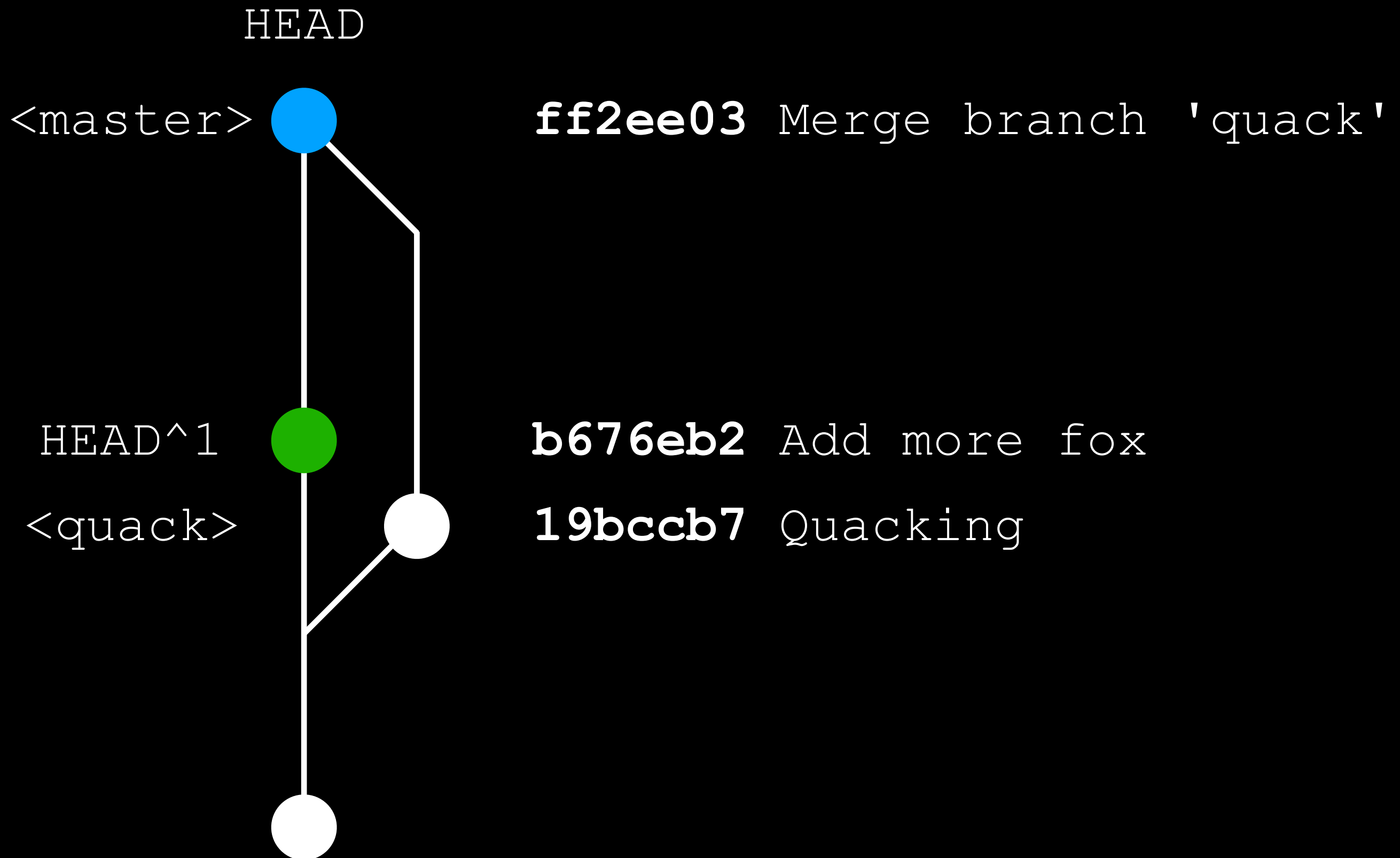
Tilde (~) and caret (^)

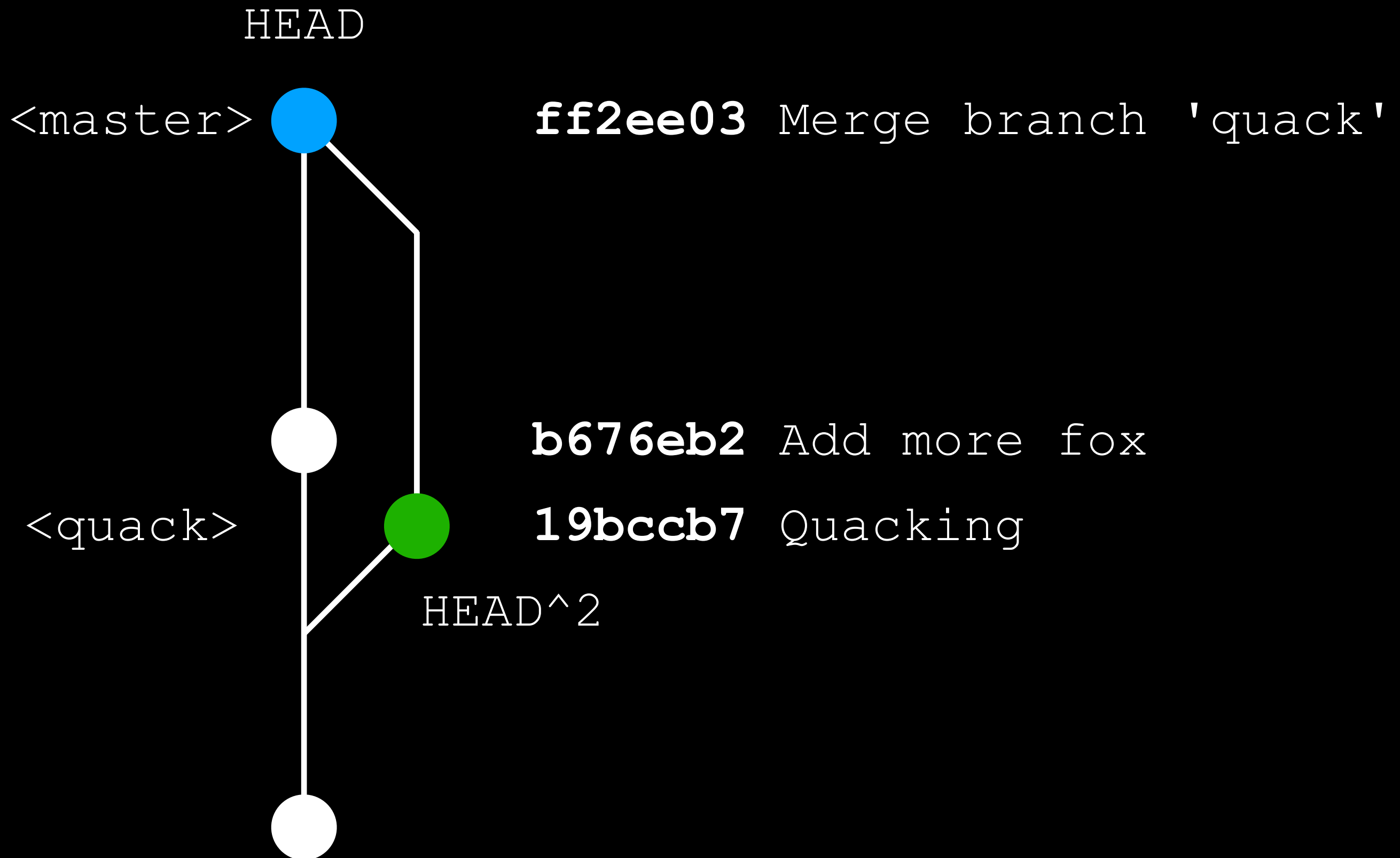
- `master~1`
- `HEAD^2`











Demo

In Summary

- `init`
- `add, commit`
- `log`
- `branch`
- `merge`
- `checkout, reset`
- `diff, show, grep`
- `master~1, HEAD^2`

Congratulations

You now know enough Git to be dangerous

- Too dangerous actually
- You may want to periodically backup your work using more traditional methods until you're more comfortable

There's a lot more to
learn

Cherry-pick

Interactive rebase

Rebasing a branch

Clean

Bisect

Creating patches

Reflog

.gitignore

Tag

Revert

Stashing changes

Blame

GitHub

Let's get started

Please get your computer ready

This part is best if you have a computer to follow along with

If you don't, no worries — you can view the slides later

The GitHub Model

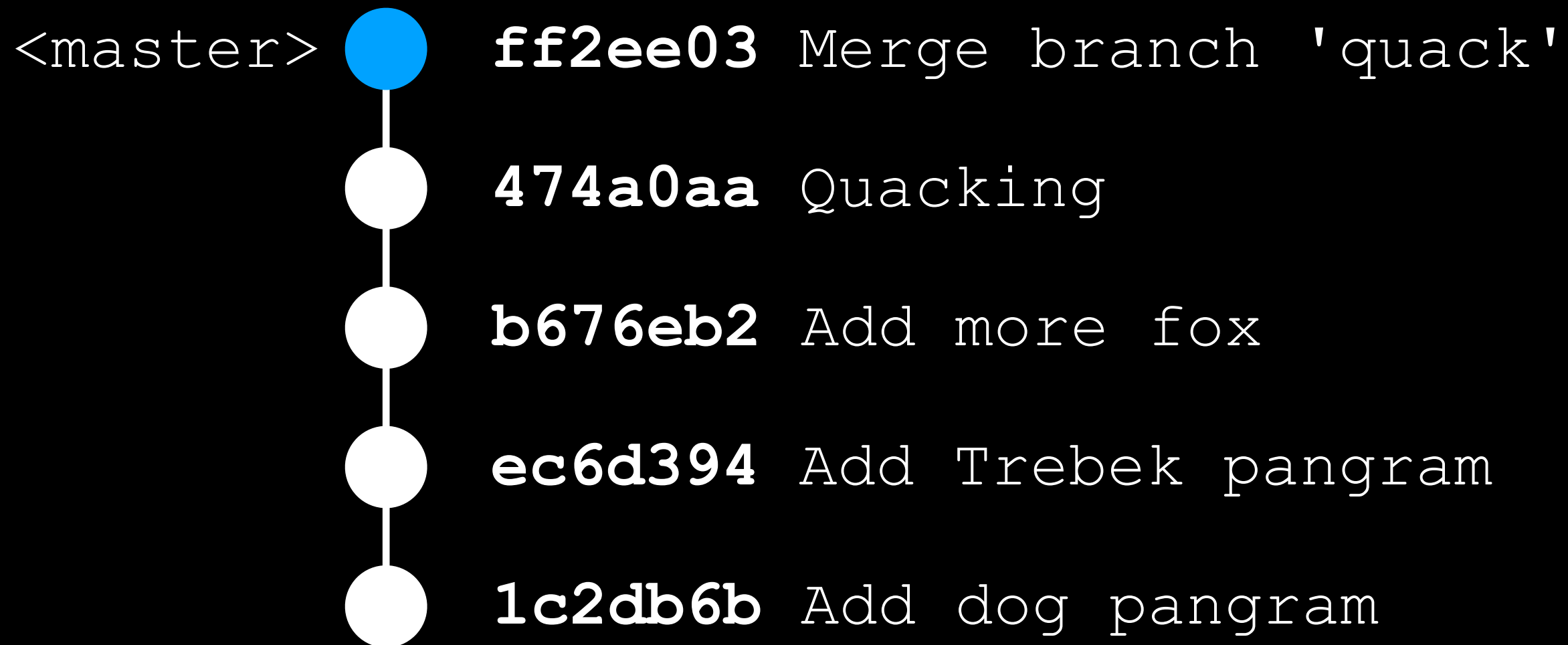
Two repos, one story

- Local repository on your machine (same as before)
- Update remote repository as desired

Push and Pull

- Commits will update local repository
- Remote repository missing changes? **Push from local**
- Updated content on remote repository? **Pull from remote**

HEAD





Local

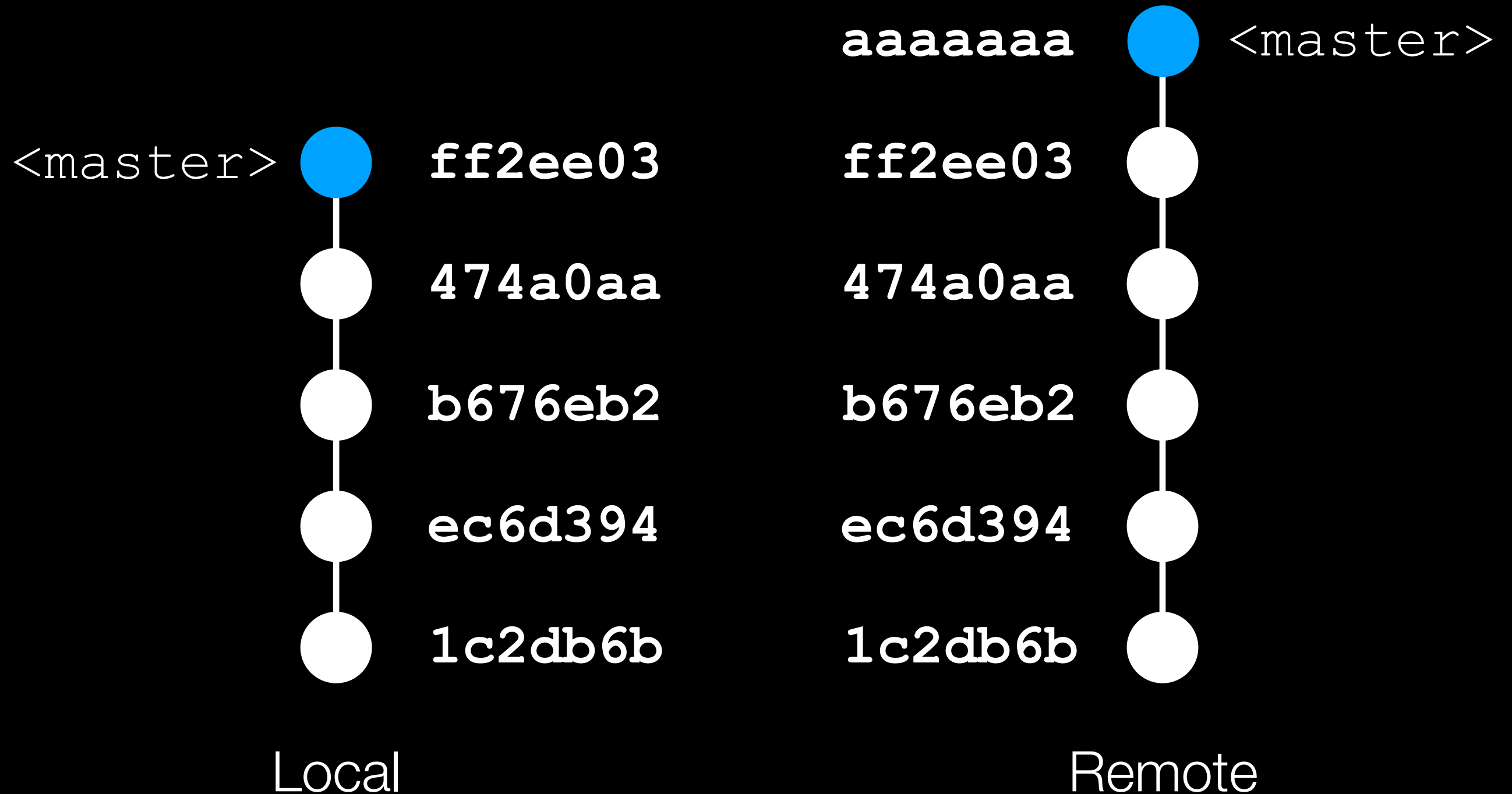
Remote

<master>

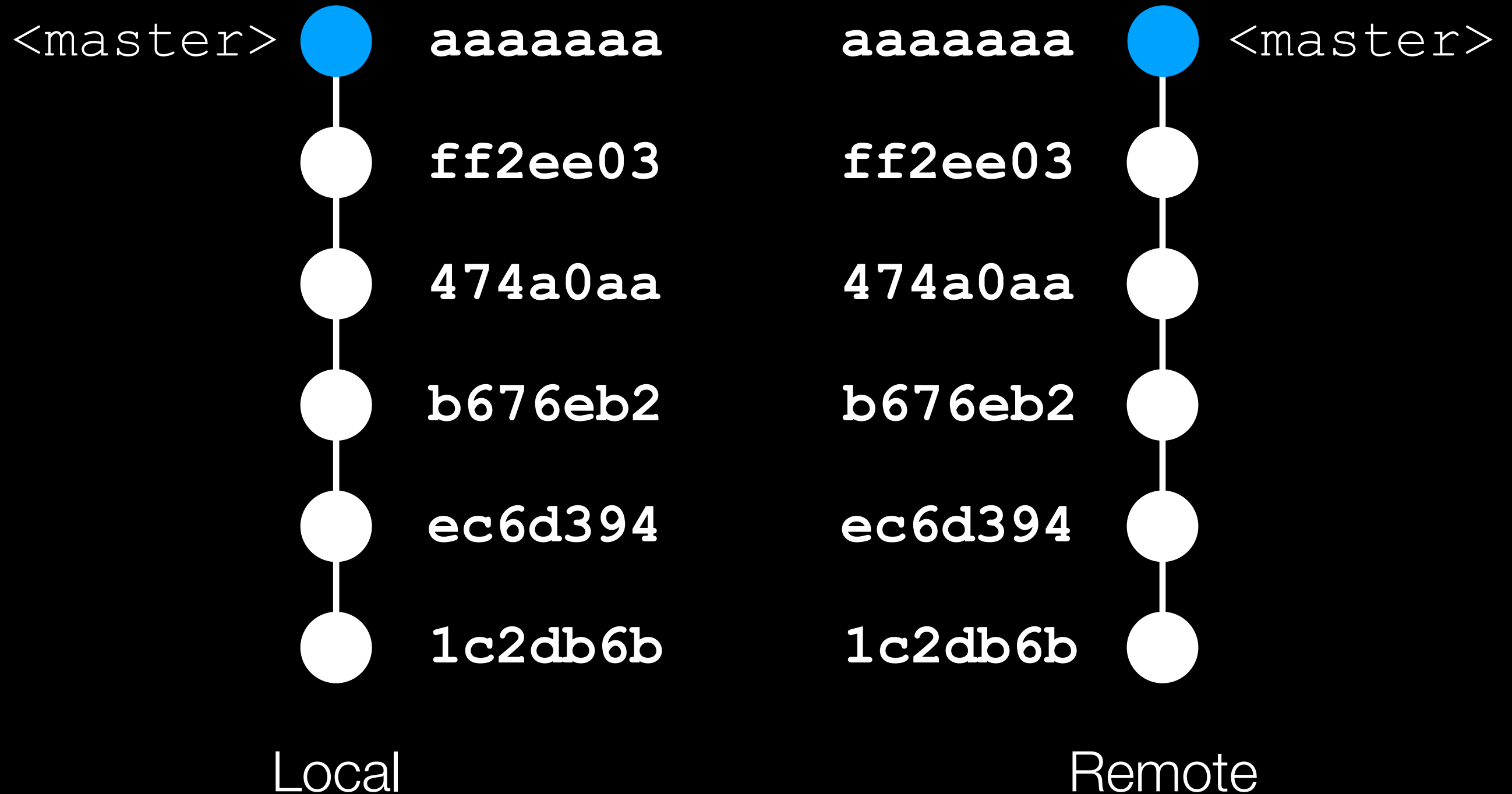
git push origin master



git commit (on another machine)



git pull origin master



Demo

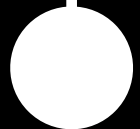
Putting it all together

Finally, the real world

- Merge conflict example from before
- Now, different people changing the same branch



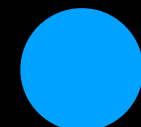
ff2ee03



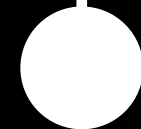
474a0aa

Jerry

Paul



ff2ee03



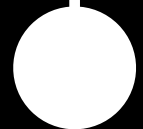
474a0aa

GitHub

git clone

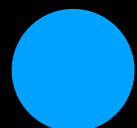


ff2ee03

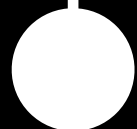


474a0aa

Jerry

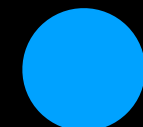


ff2ee03

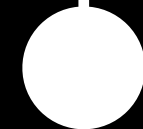


474a0aa

Paul



ff2ee03

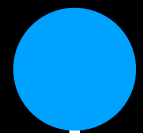


474a0aa

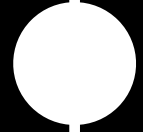
GitHub

git commit

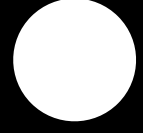
git commit



1111111

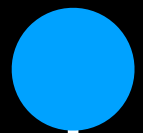


ff2ee03

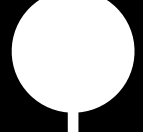


474a0aa

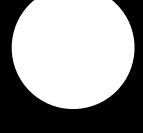
Jerry



aaaaaaaa

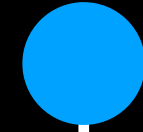


ff2ee03

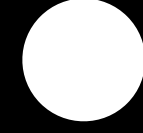


474a0aa

Paul



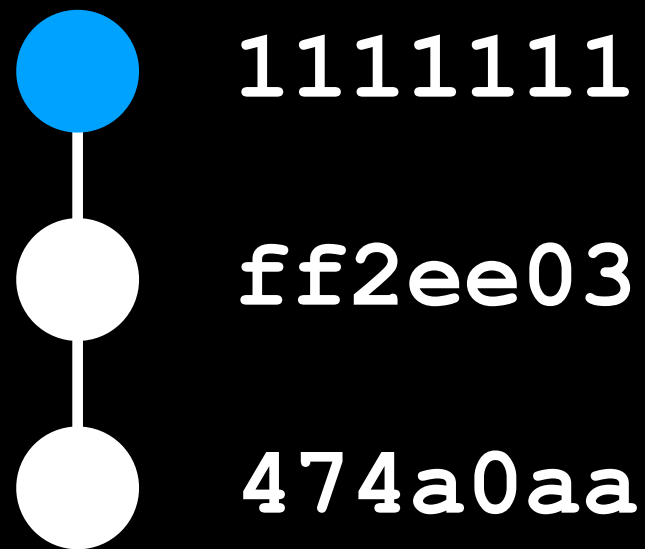
ff2ee03



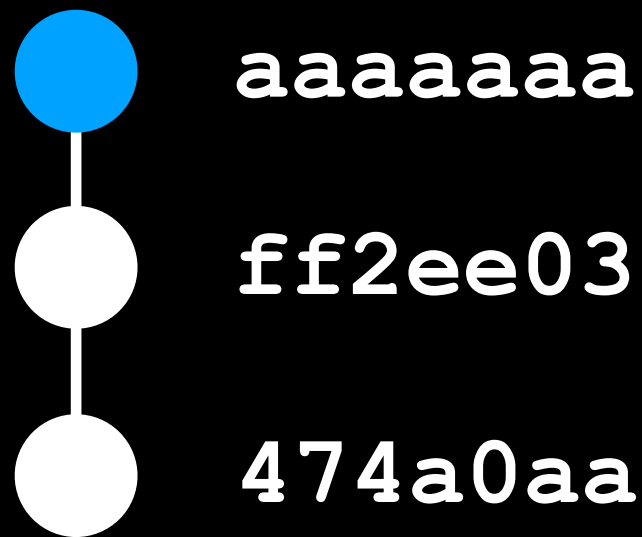
474a0aa

GitHub

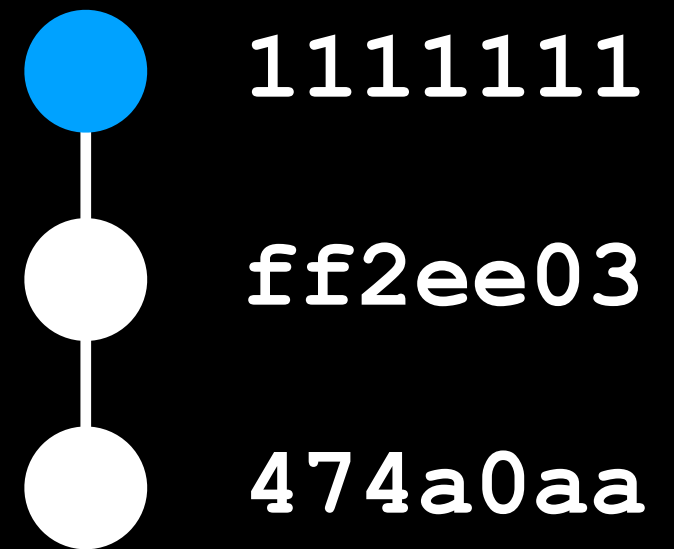
git push



Jerry

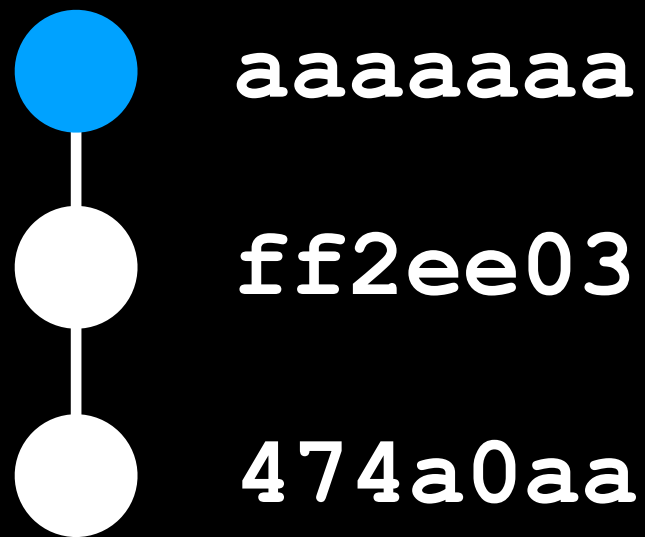


Paul

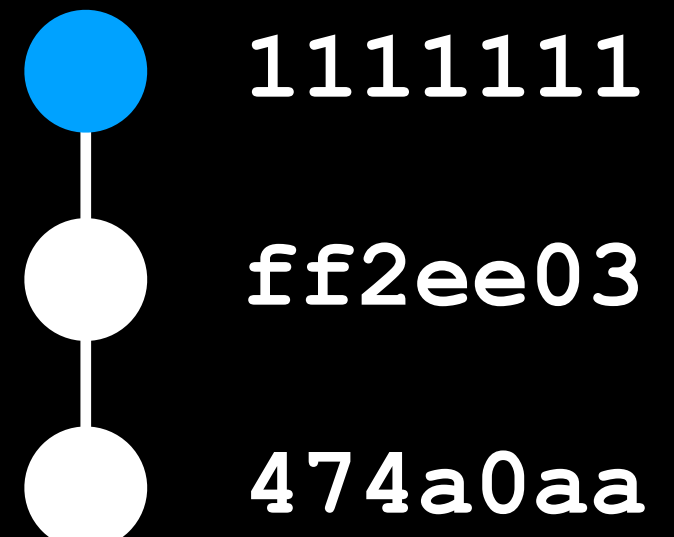


GitHub

 git push

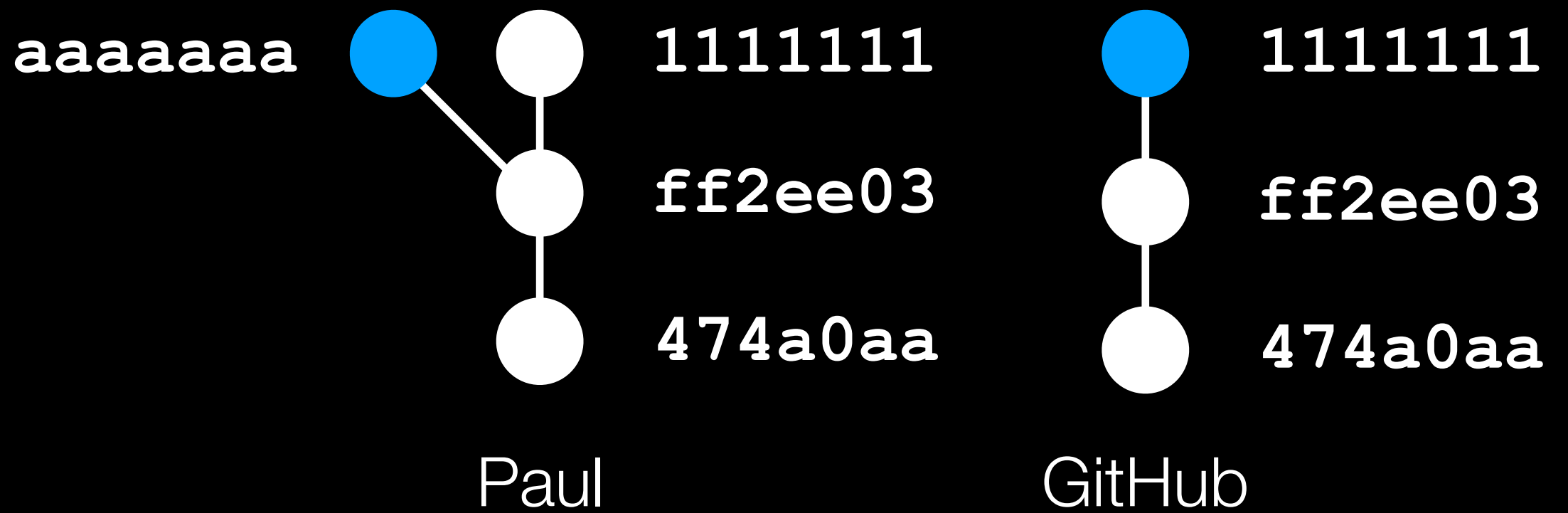


Paul

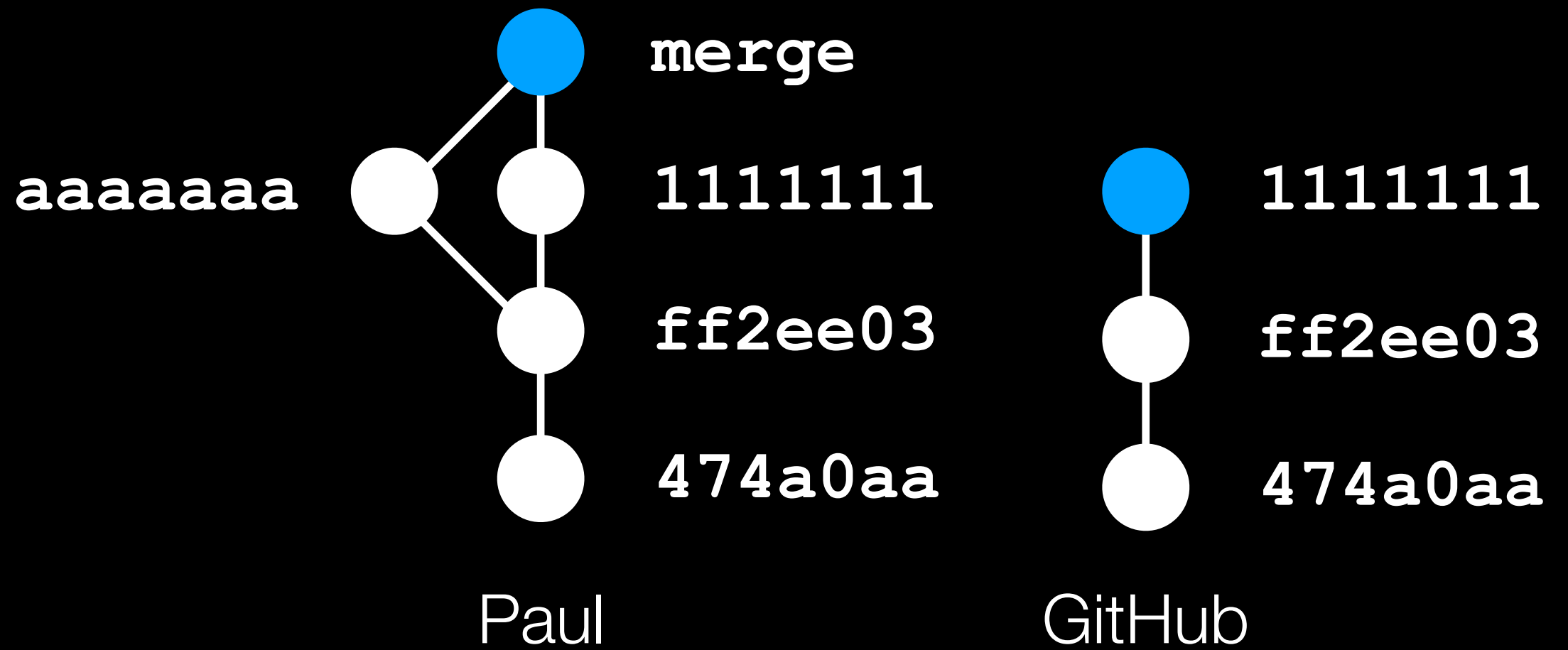


GitHub

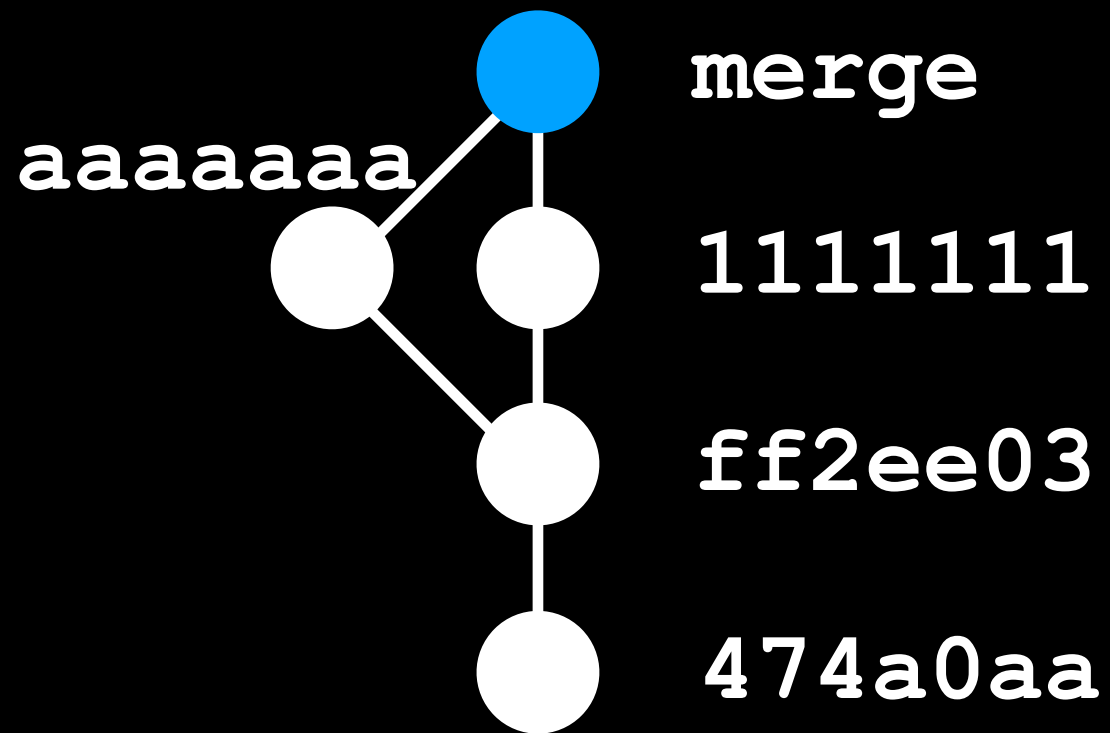
git pull



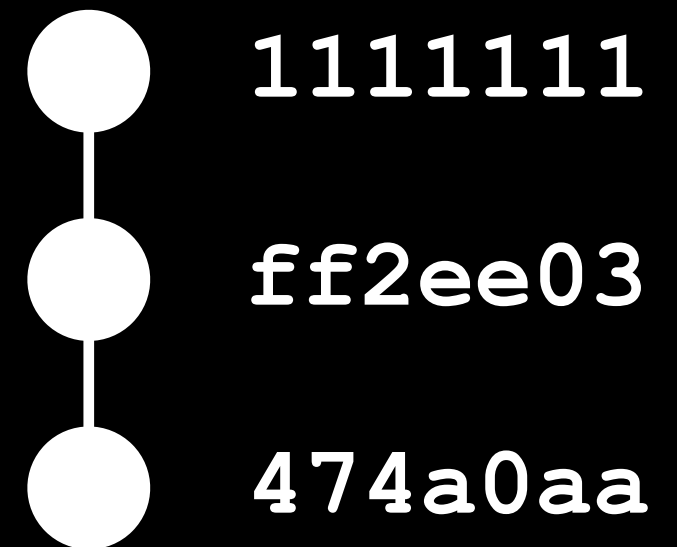
git pull



git push

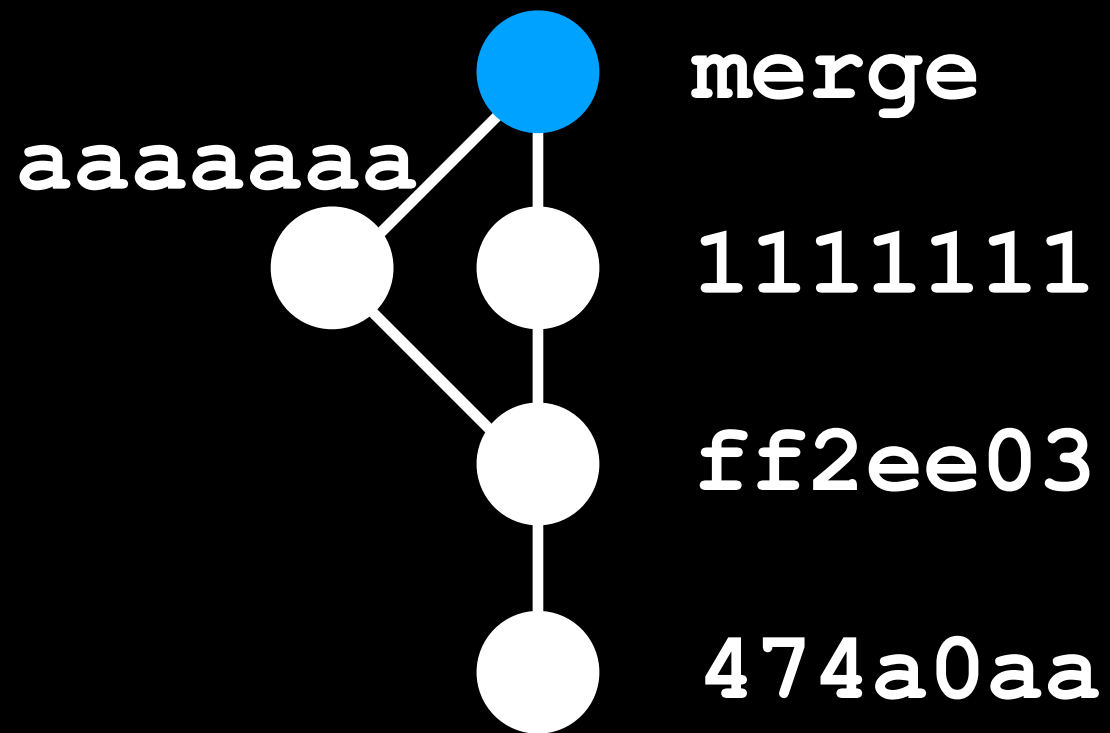


Paul

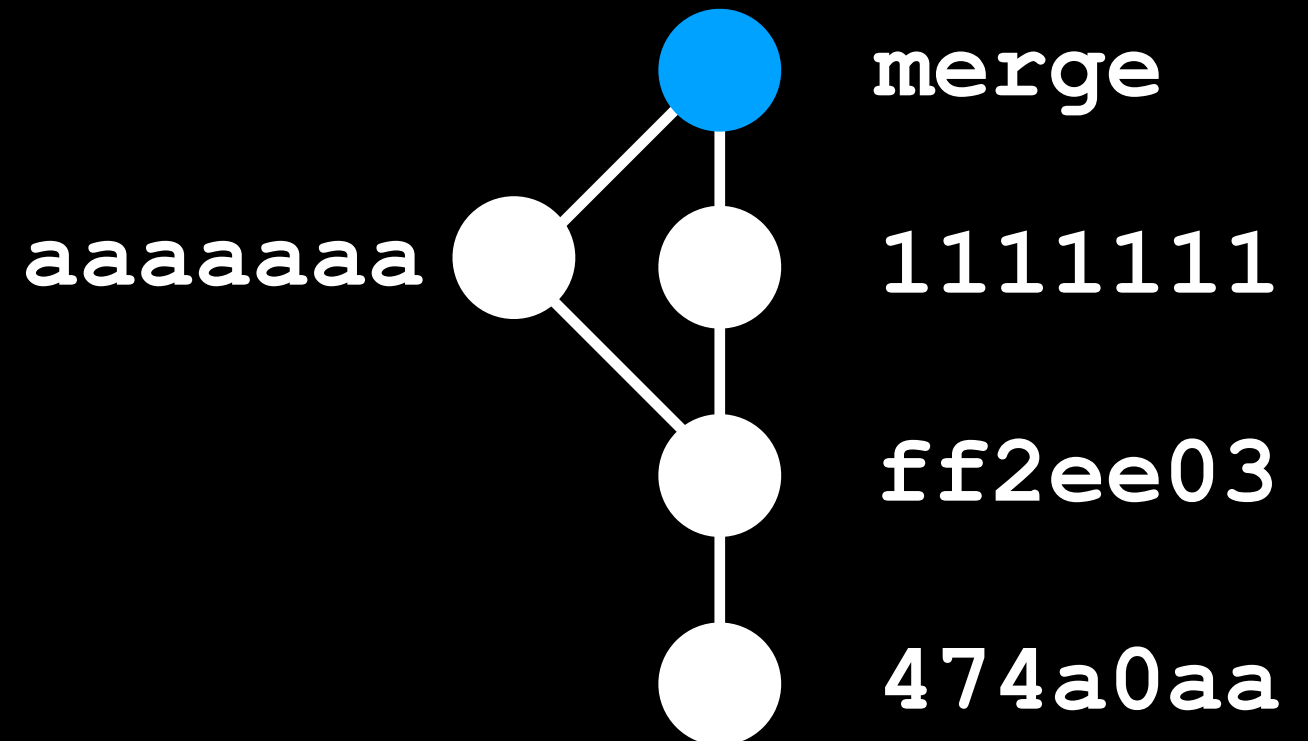


GitHub

git push



Paul



GitHub

Demo

Projects

GitLab

Signatures

Pull requests

Fork

Fetch

Issues

Organizations

Customizing remotes

ssh keys

Bitbucket

Reference Material

Miscellaneous

Additional Resources

- GitHub's own tutorial: guides.github.com/activities/hello-world/
- Popular Git GUIs:
 - Git Tower: git-tower.com
 - SourceTree: source-treeapp.com
- Digital signatures for commits: <https://help.github.com/articles/signing-commits-using-gpg/>
- Git plugin for ohmyzsh (also install themes with Git goodies): <https://github.com/robbyrussell/oh-my-zsh/blob/master/plugins/git/git.plugin.zsh>
- Writing good commit messages: <https://chris.beams.io/posts/git-commit/>

```
git commit -m  
"End of workshop"
```