Version Control with Git

02.05.2016

Max Schelker
Max Flöttmann

Motivation

Alice

```
print("Hello Alice!")
```

```
def sayHello():
    print("Hello Alice!")

sayHello()
```

```
name = input("What's your name?\n")
def sayHello(name):
    print('\nHello '+name+'!\n')

sayHello(name)
```

Bob

```
print("Hello Bob!")
```

```
isbob = input("Is it Bob?\n”)
if (isbob == "y"):
    print("\nHello Bob!\n")
```

```
isbob = input("Is it Bob?\n”)
if (isbob.lower() == "y"):
    print("\nHello Bob!\n")
else:
    print("\nHello stranger!\n")
```

Motivation

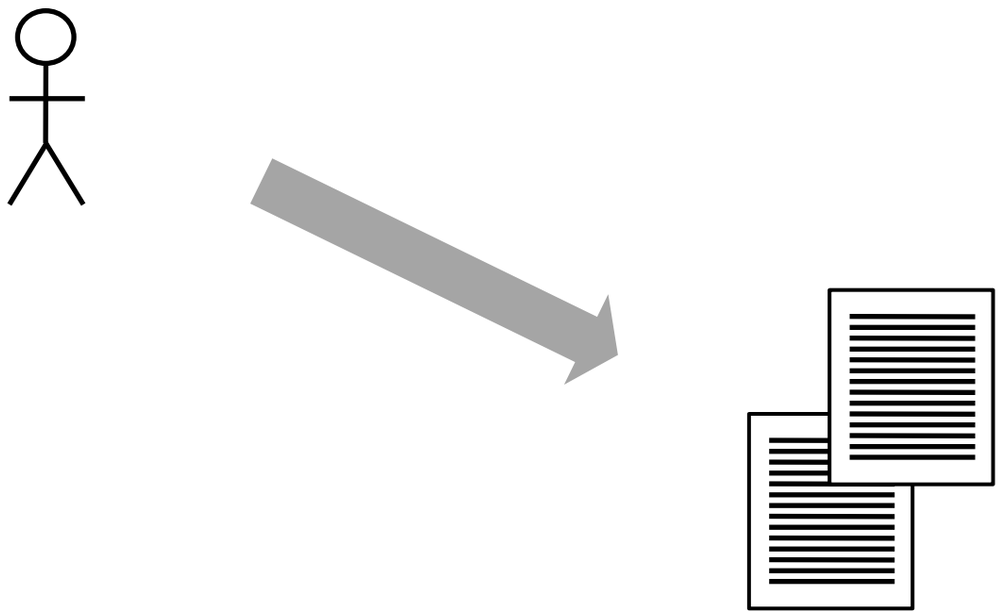
Alice

Bob

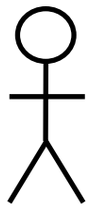
```
isbob = input("Is it Bob?\n”)
if (isbob.lower() == "y"):
    print("\nHello Bob!\n")
else:
    name = input("What's your name?\n")
    print('\nHello '+name+'!\n')
```

Why git?

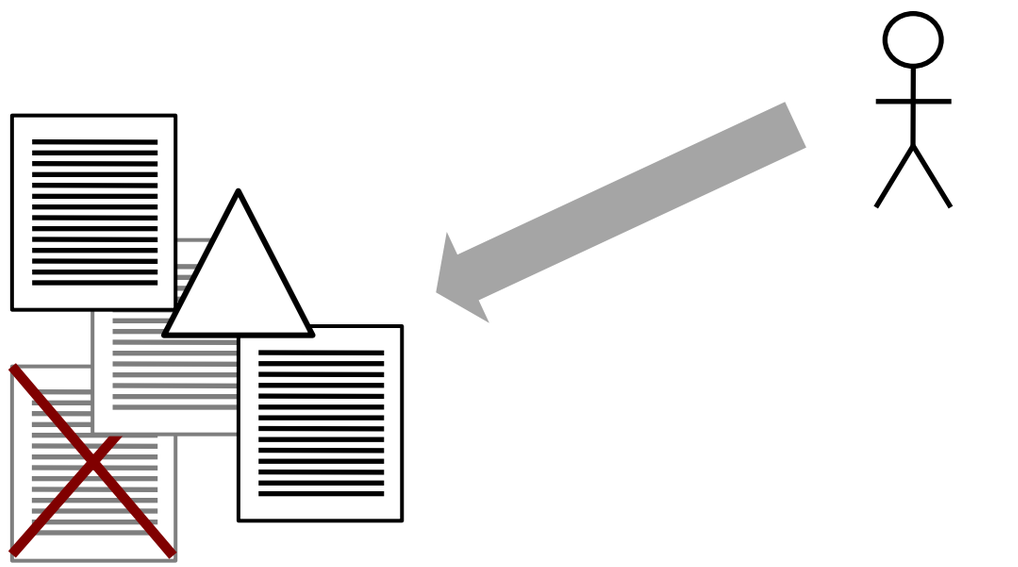
Bob



Bob

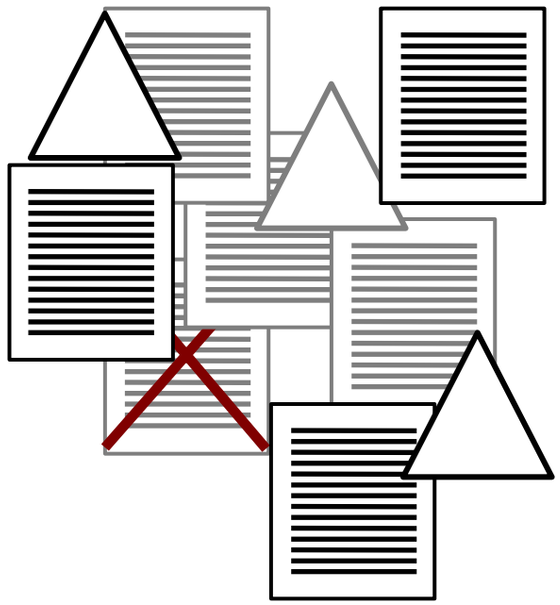


Carol



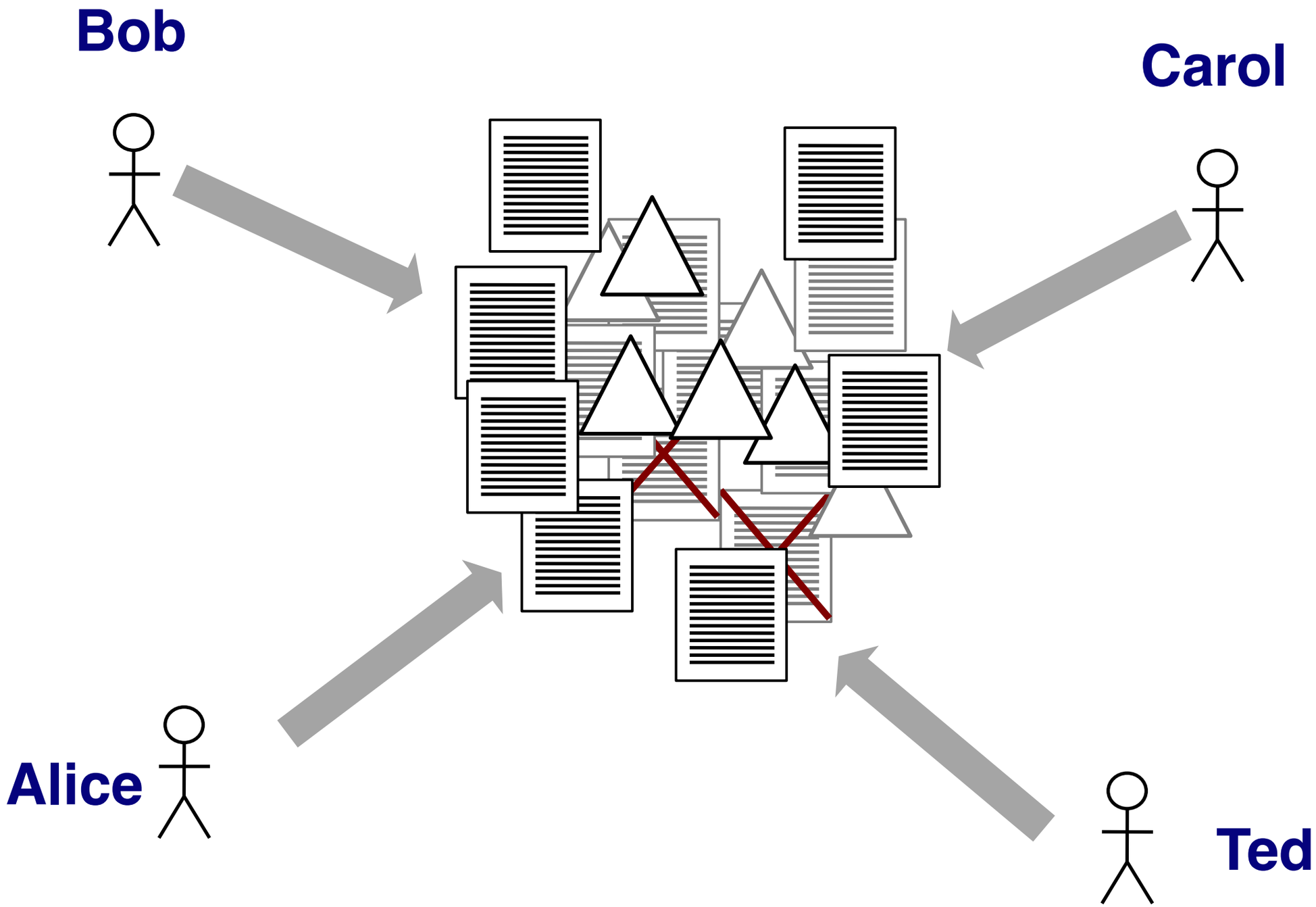
Bob

Carol



Alice

Ted



A recipe for disaster!

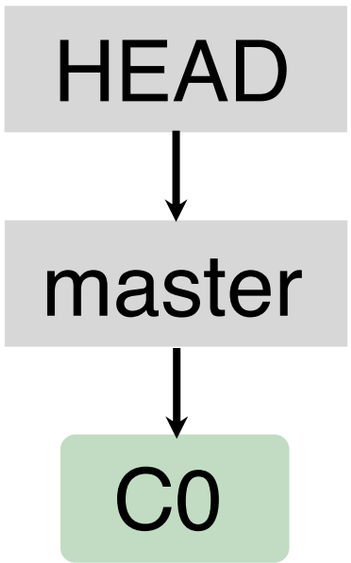
What is git?

- Distributed revision control system
- Originally developed by Linus Torvalds for the development of the Linux Kernel in 2005
- Focus on speed and efficiency
- Quite a unique design and therefore sometimes a bit scary and difficult to understand

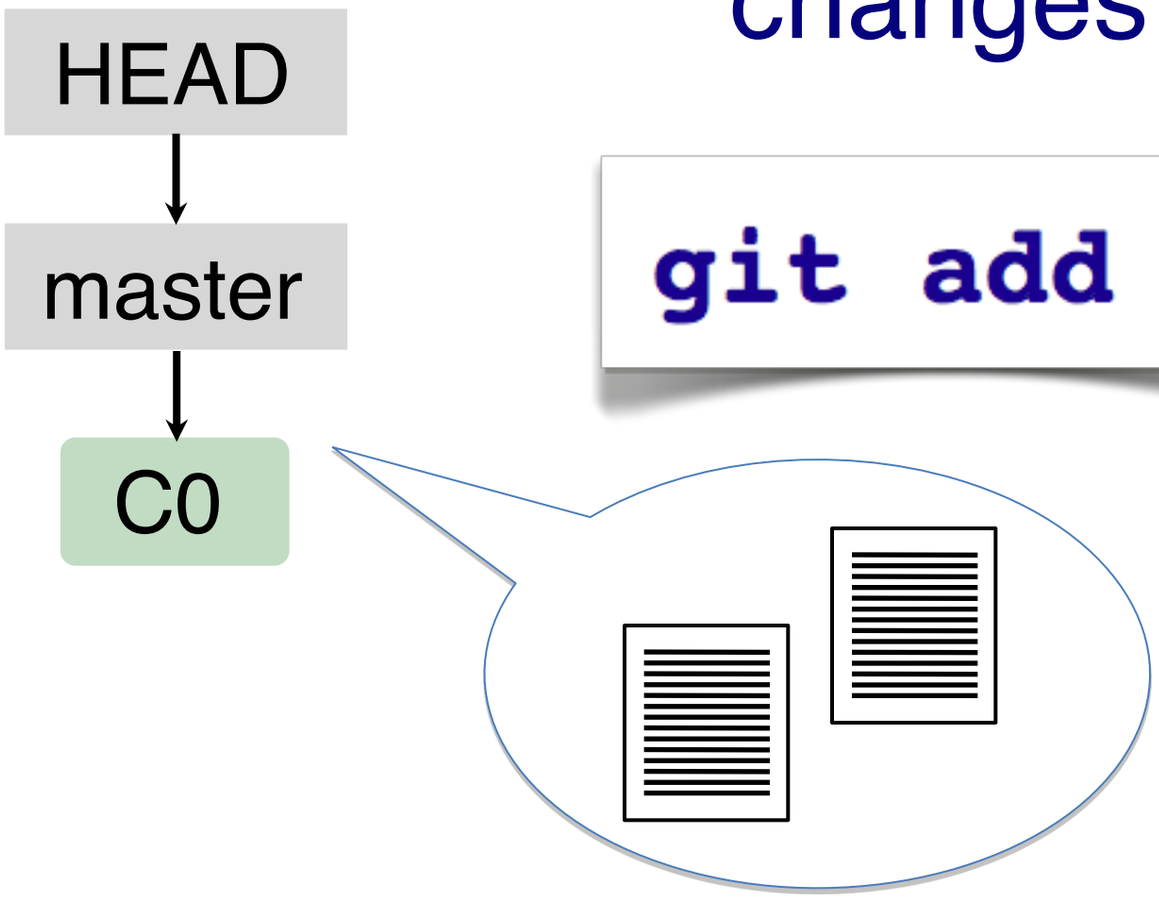
Basic git

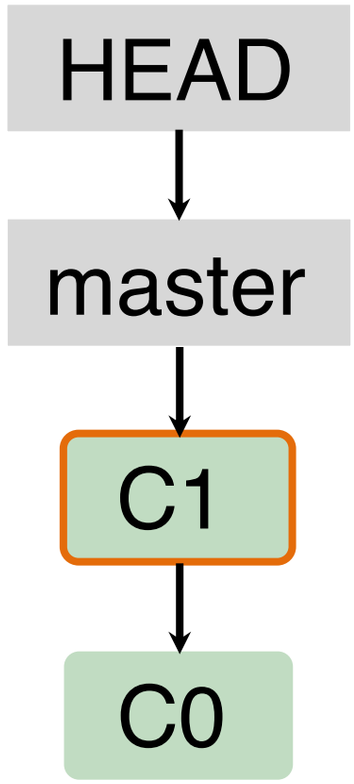
Create a git repo

```
mkdir repo
cd repo
git init
```



Tell git to “stage” changes





Commit your changes

```
git commit …
```

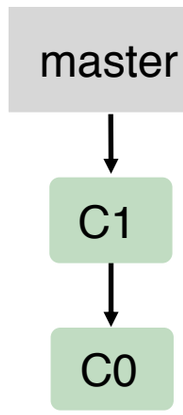
Collaborating

John

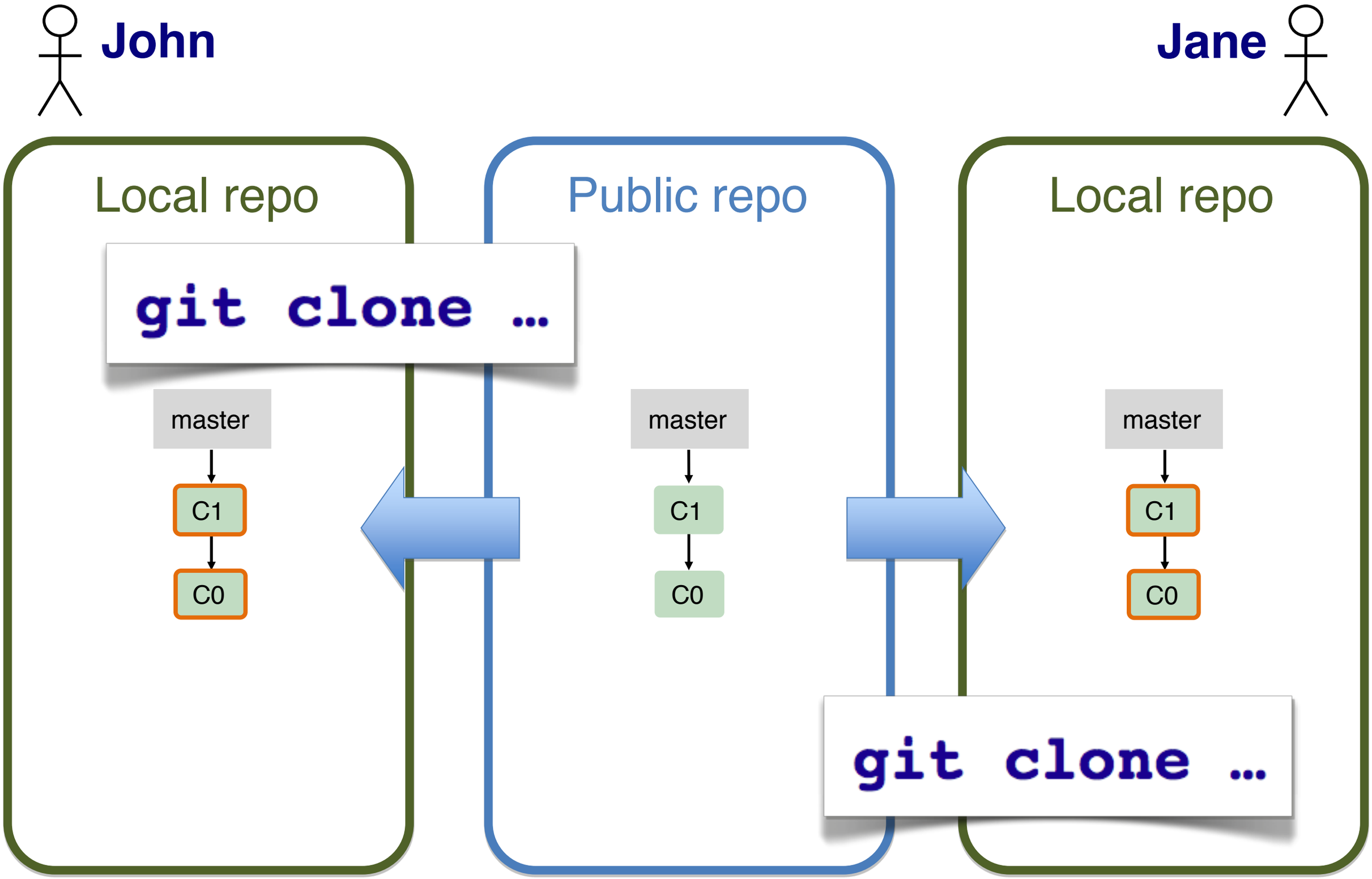
Jane

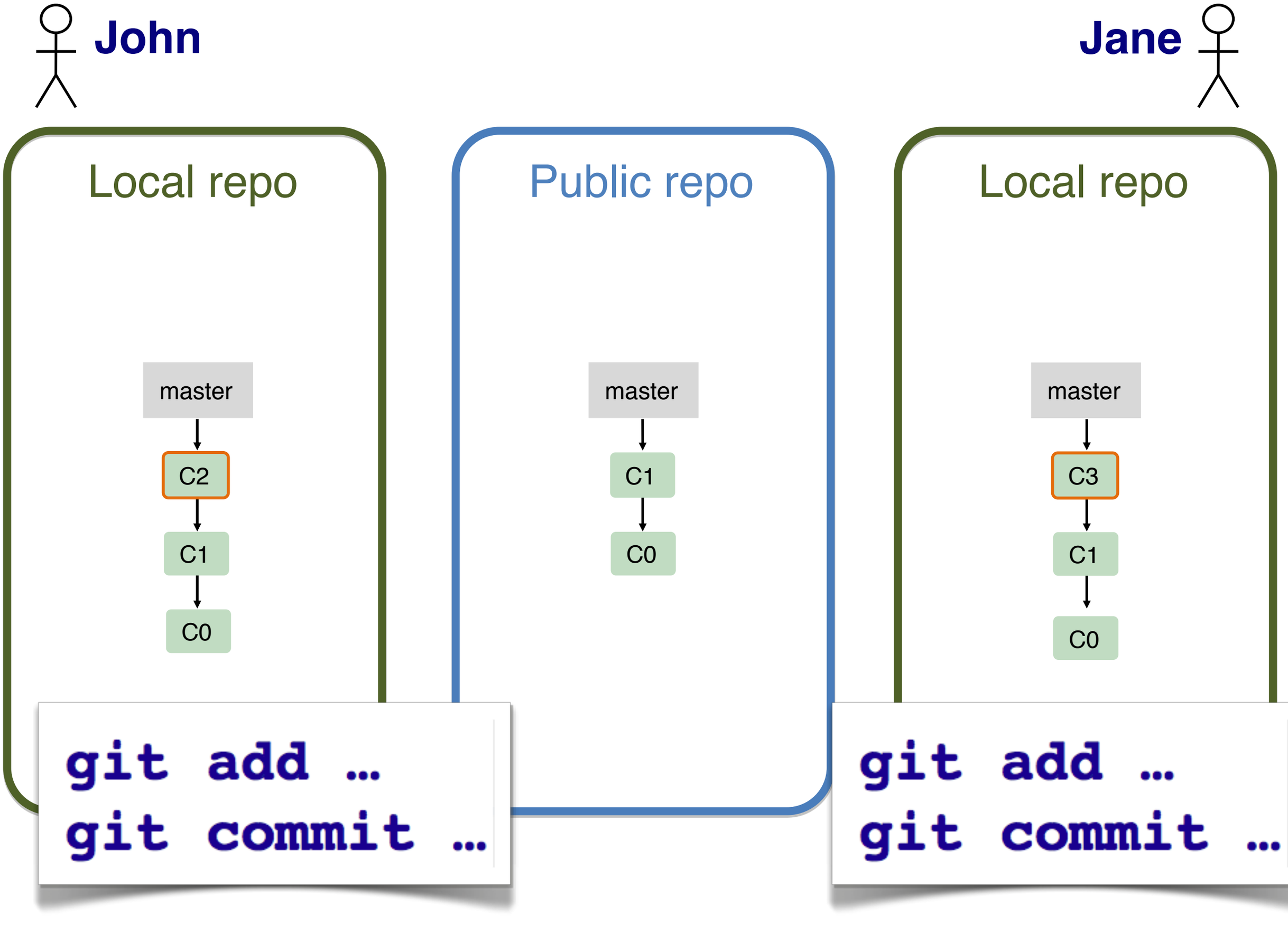
Local repo

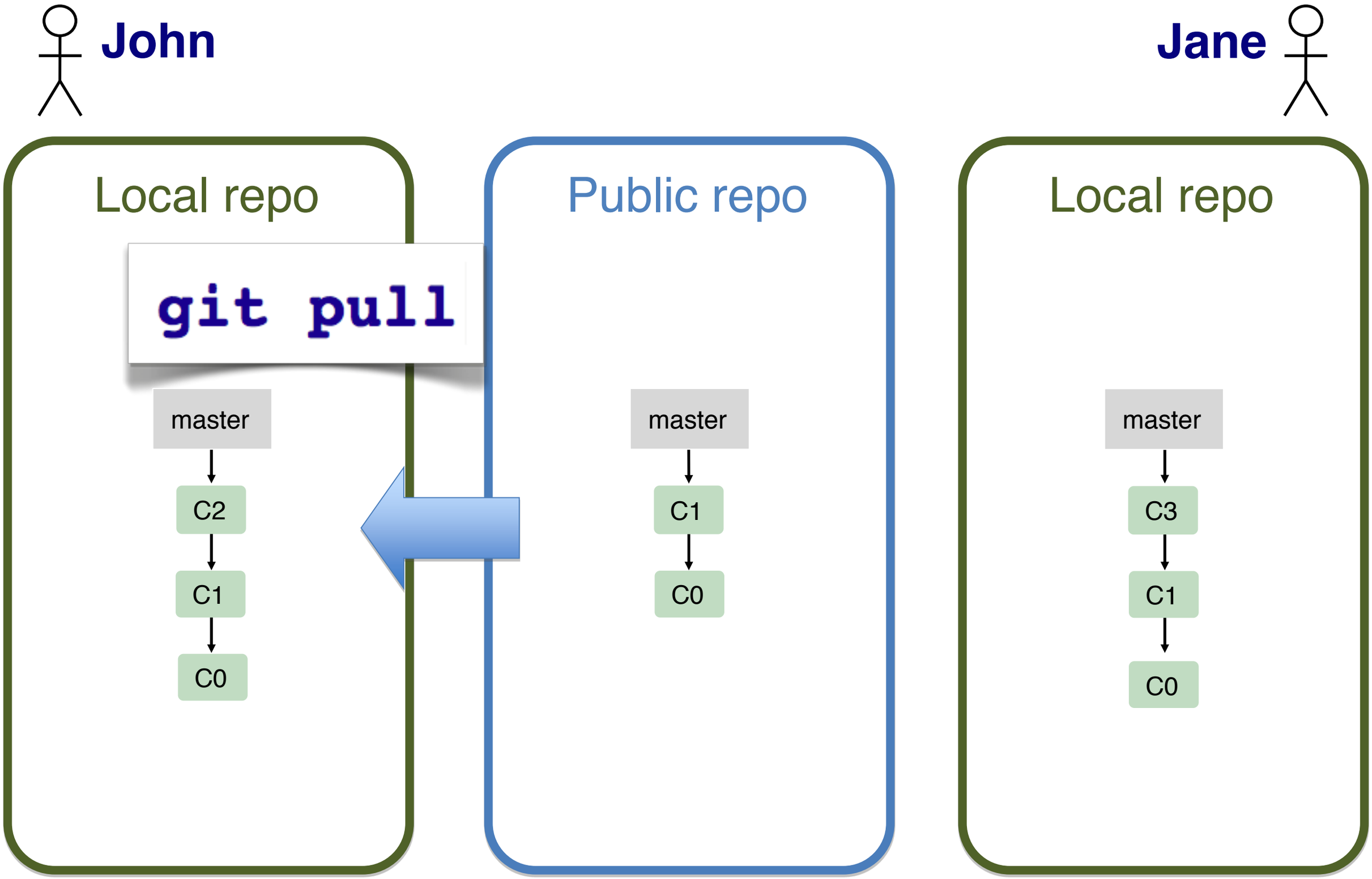
Public repo



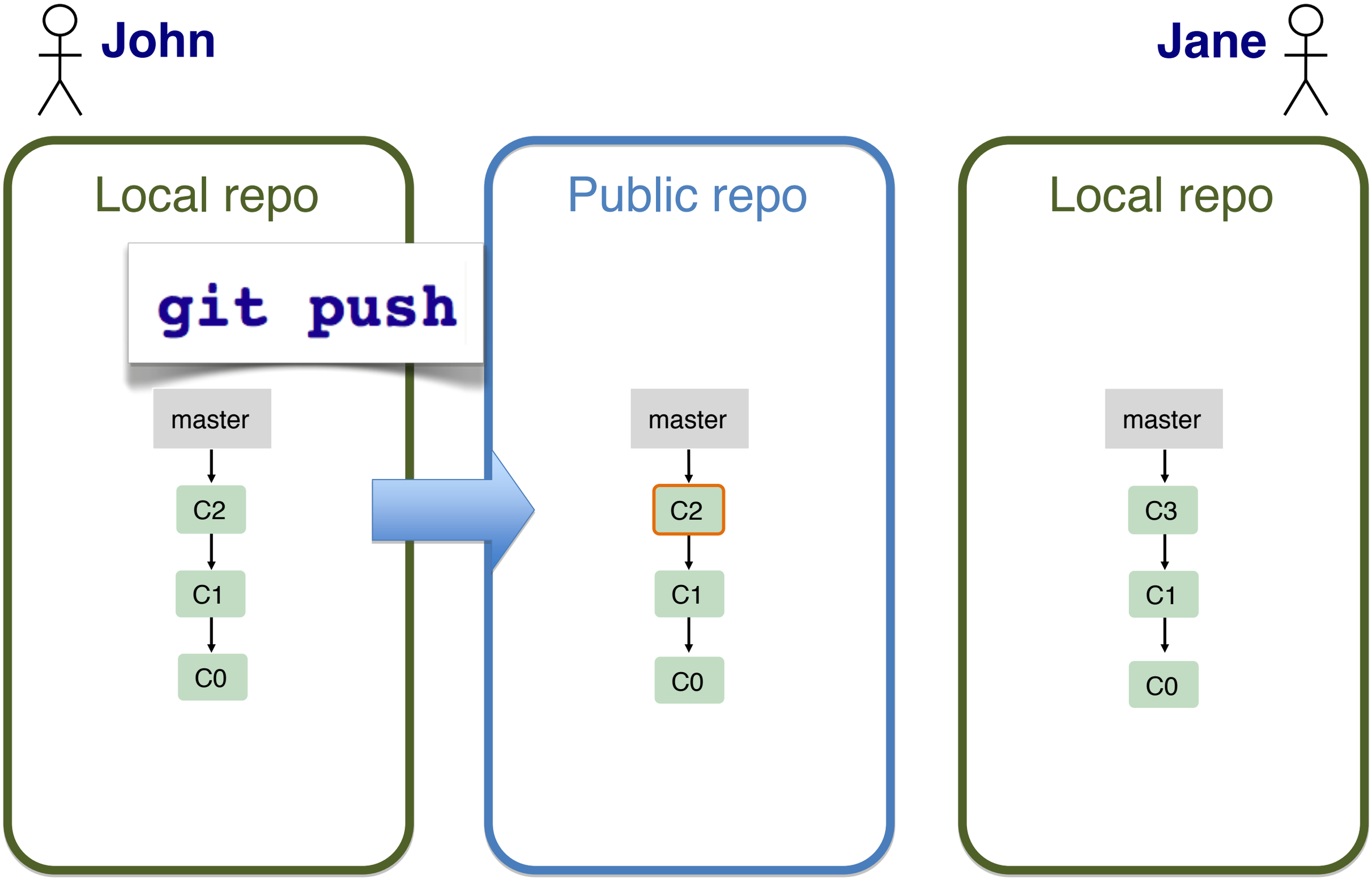
Local repo

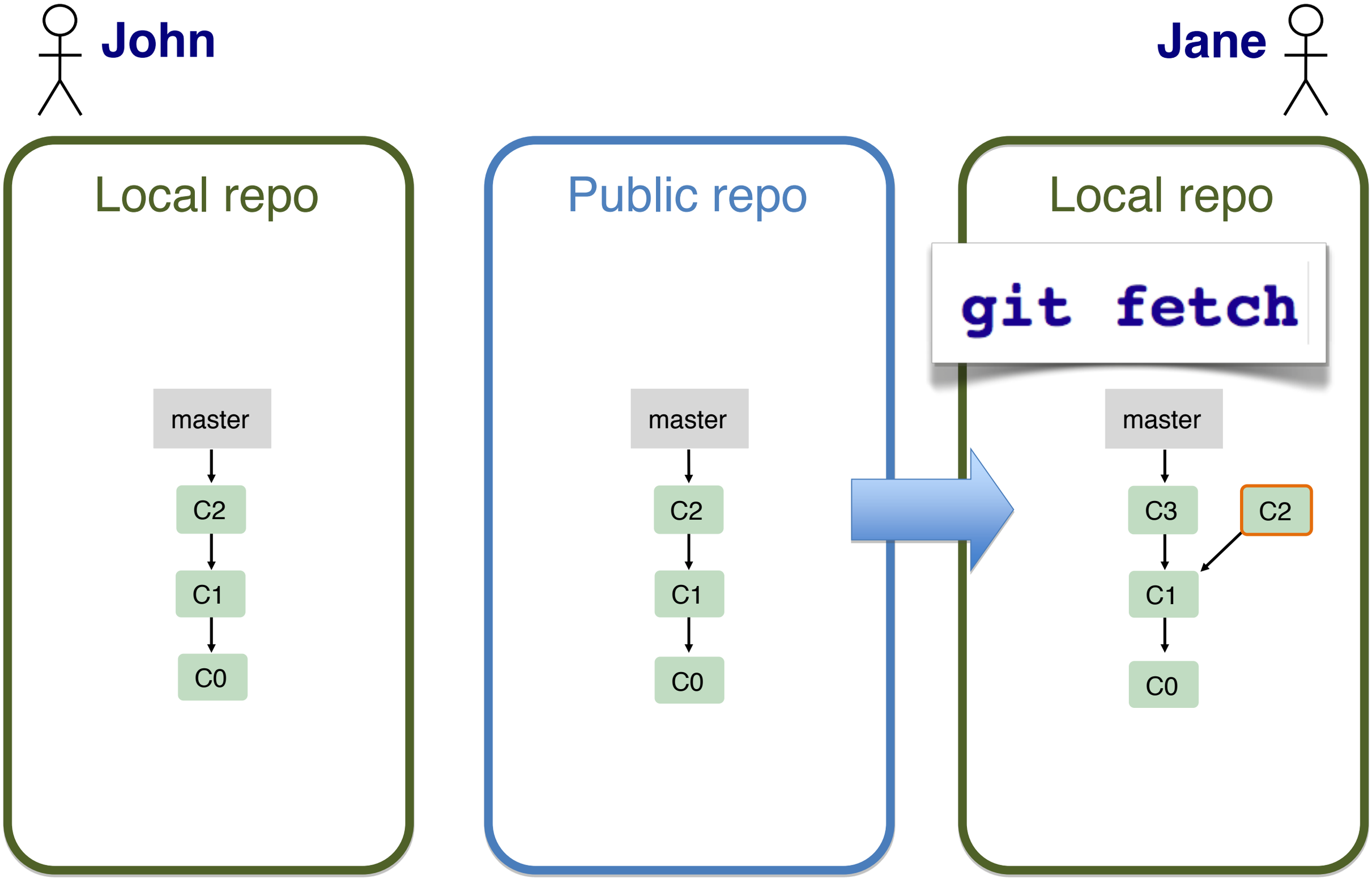

John
Jane
Local repo
Public repo
Local repo
git clone …
master
C1
C0
master
C1
C0
master
C1
C0
git clone …

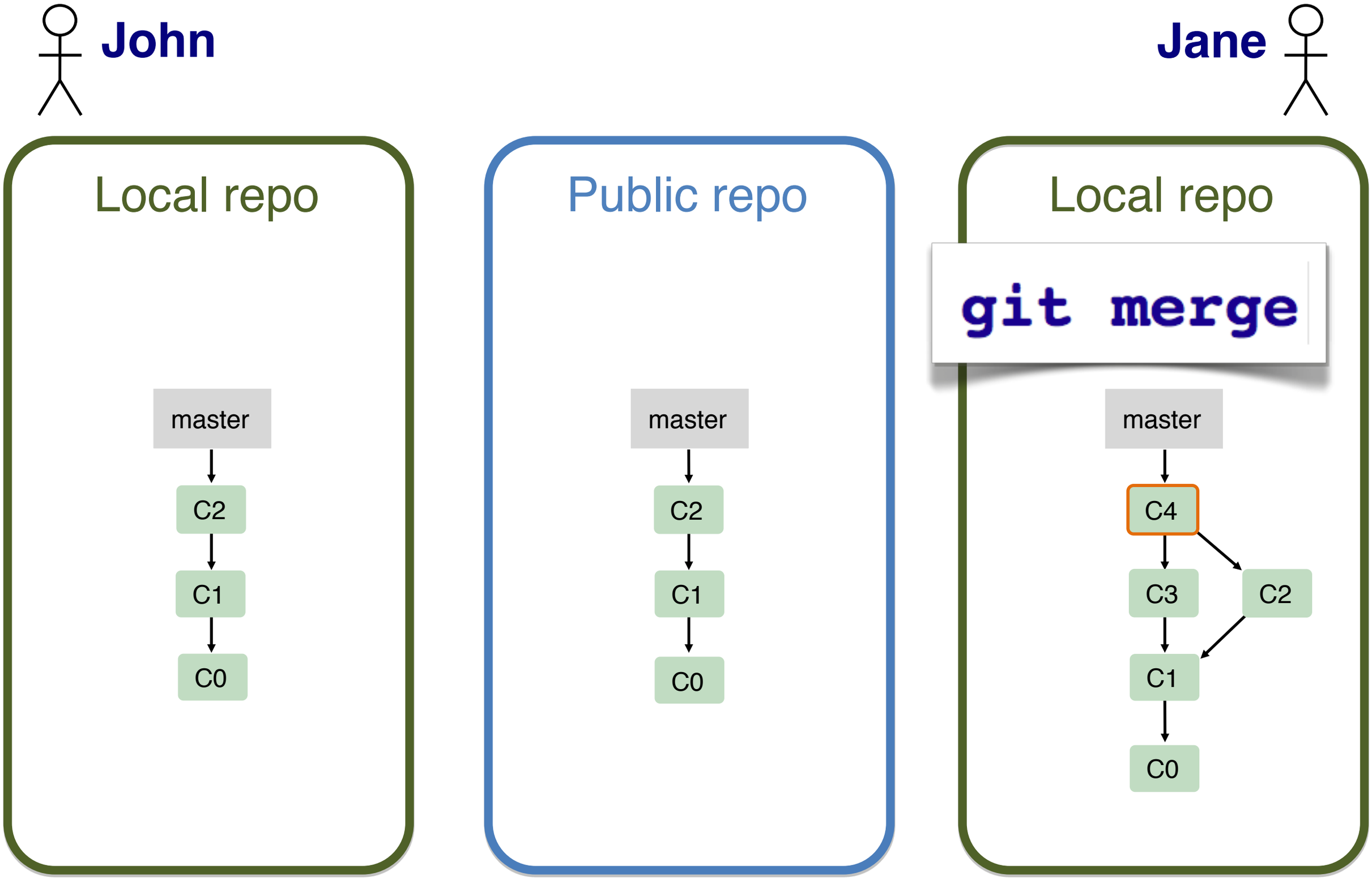

John
Jane
Local repo
Public repo
Local repo
master
C2
C1
C0
master
C1
C0
master
C3
C1
C0
git add …
git commit …
git add …
git commit …



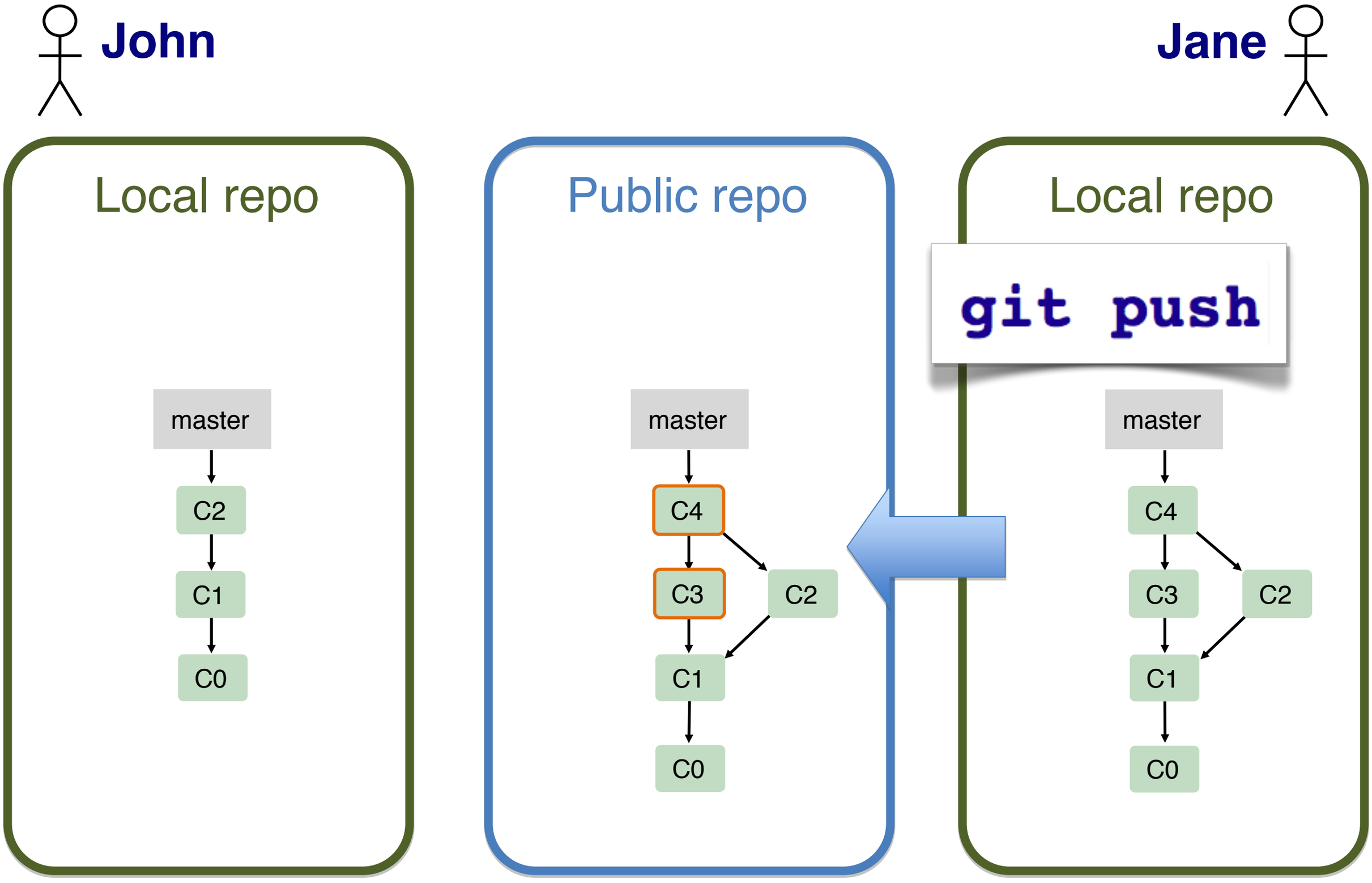
(nothing new to pull)

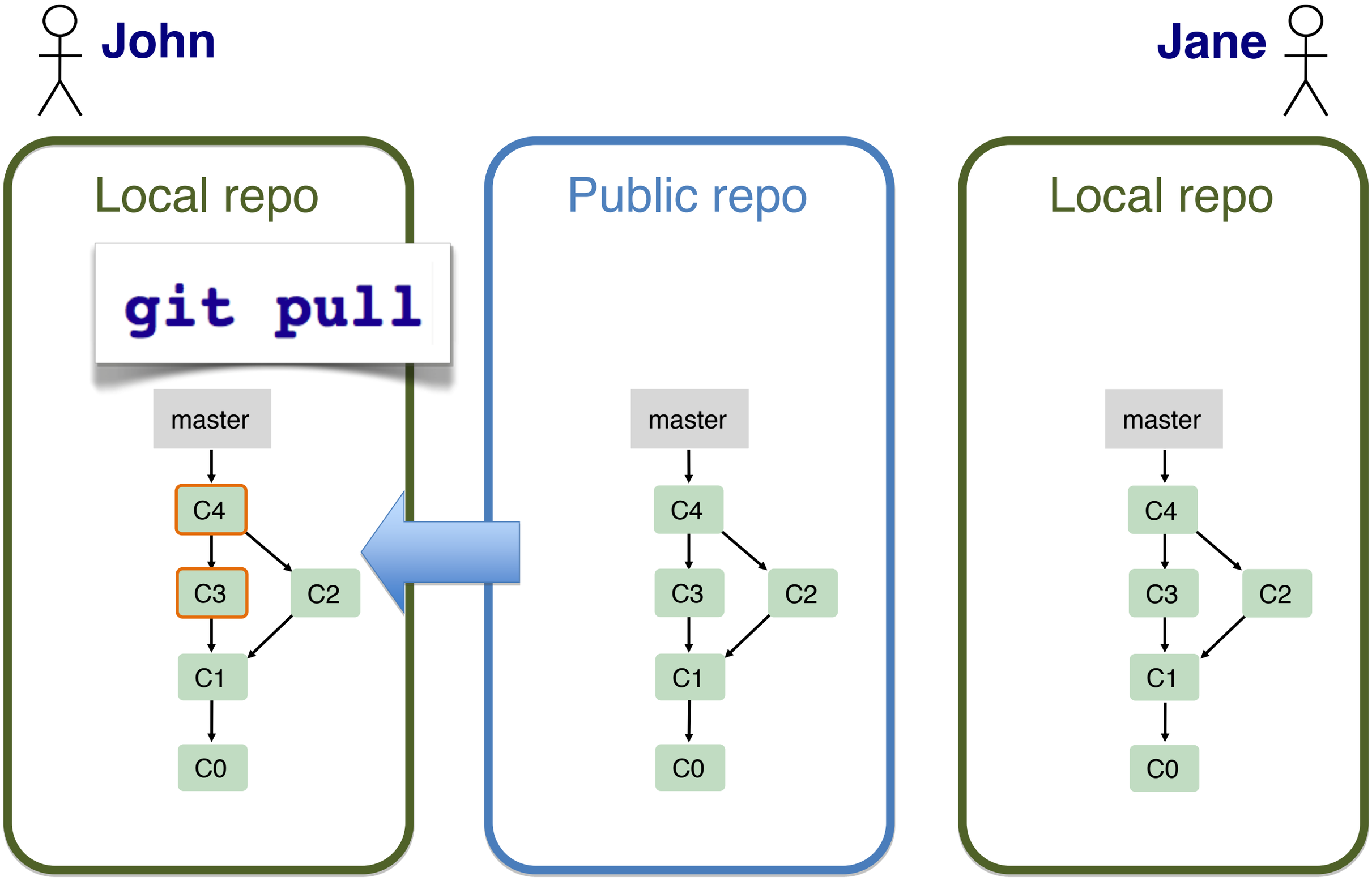

John
Jane
Local repo
Public repo
Local repo
git push
master
C2
C1
C0
master
C2
C1
C0
master
C3
C1
C0


John
Jane
Local repo
Public repo
Local repo
git fetch
master
C2
C1
C0
master
C2
C1
C0
master
C3
C2
C1
C0



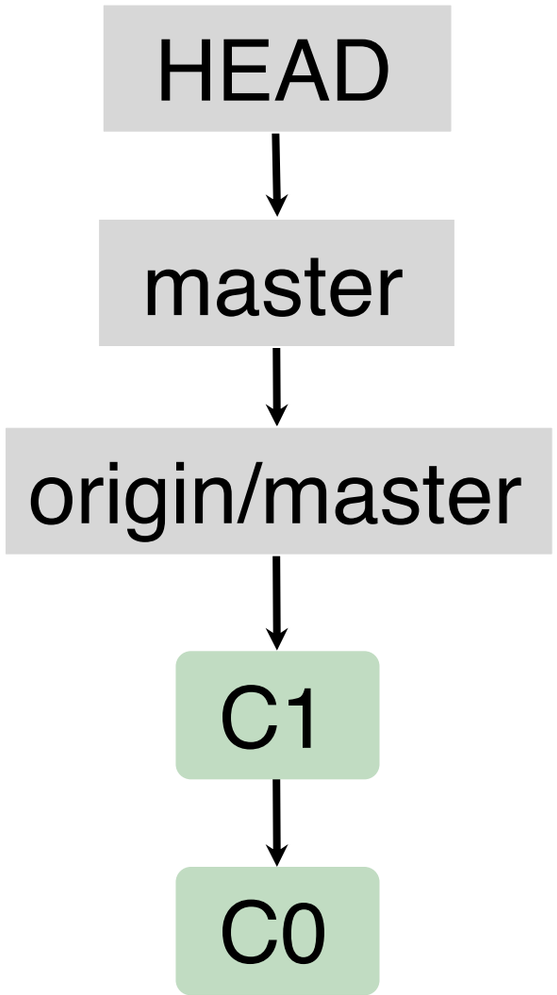
NB: **git pull** = fetch + merge

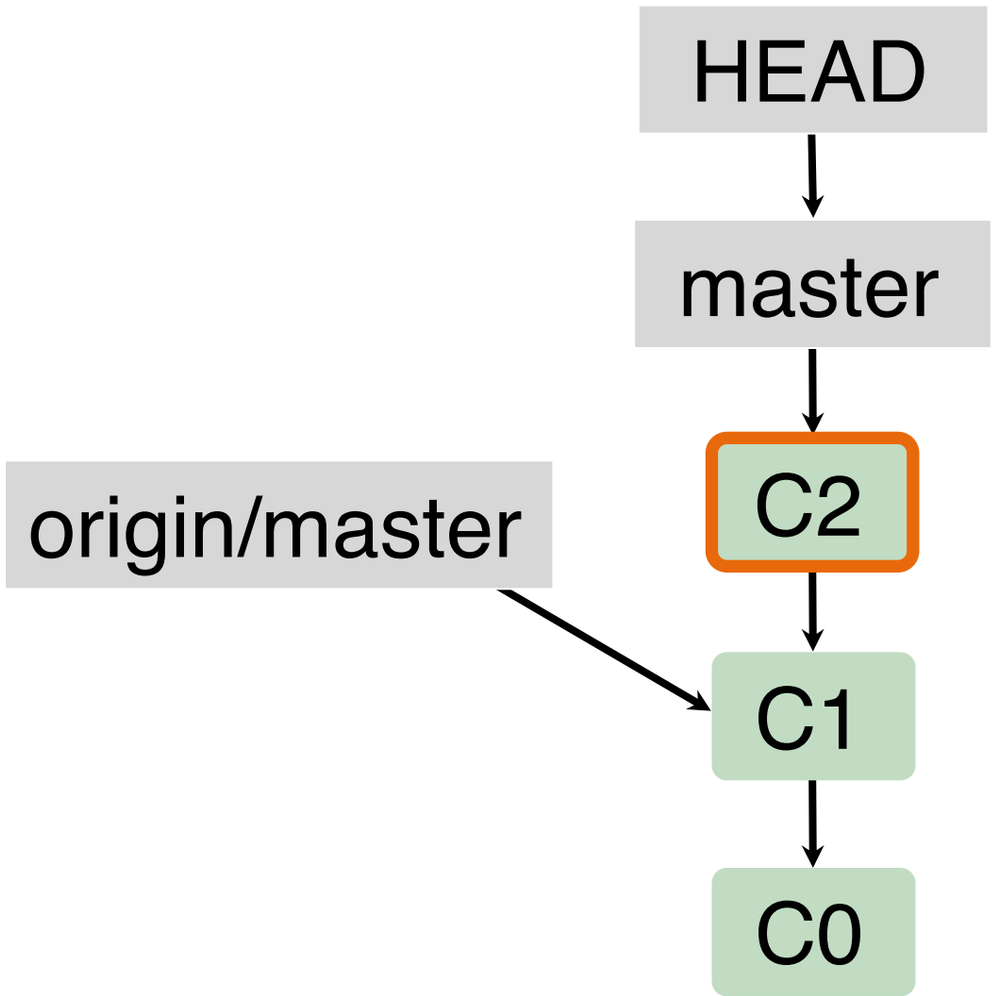

John
Jane
Local repo
Public repo
Local repo
git push
master
C2
C1
C0
master
C4
C3
C2
C1
C0
master
C4
C3
C2
C1
C0


John
Jane
Local repo
Public repo
Local repo
git pull
master
C4
C3
C2
C1
C0
master
C4
C3
C2
C1
C0
master
C4
C3
C2
C1
C0

Branching and merging

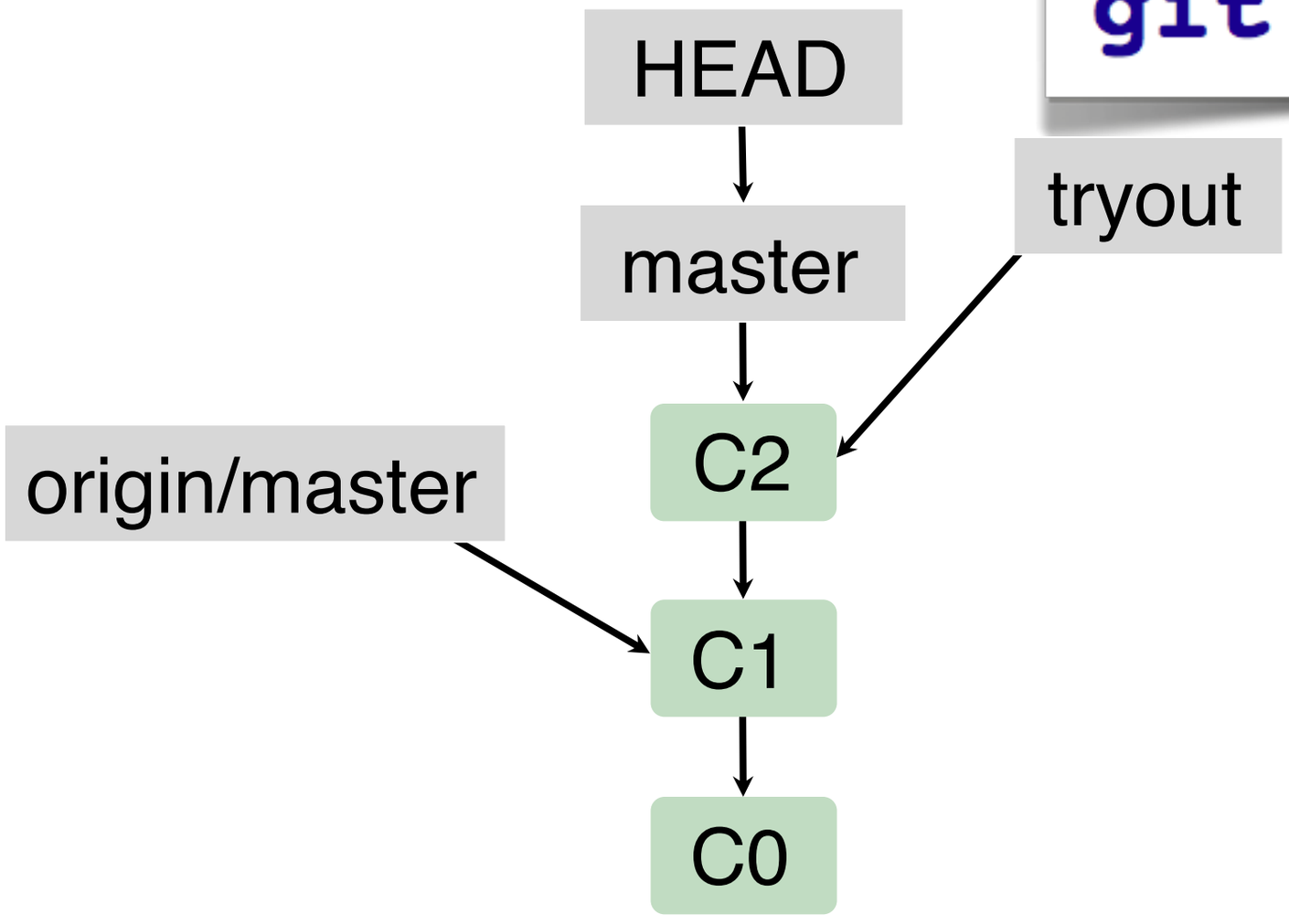
“origin” refers to the remote repo



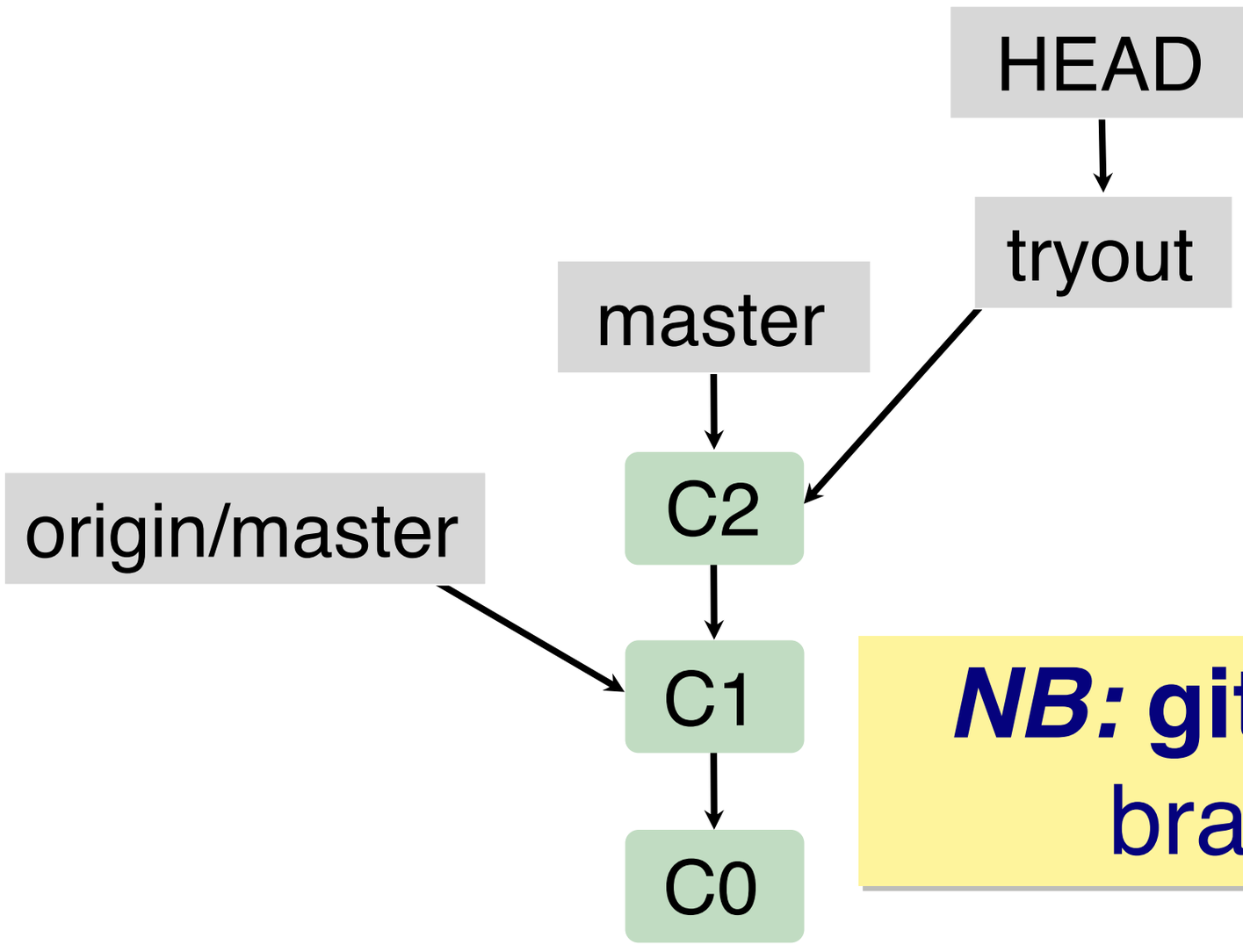



```
…
git commit …
```

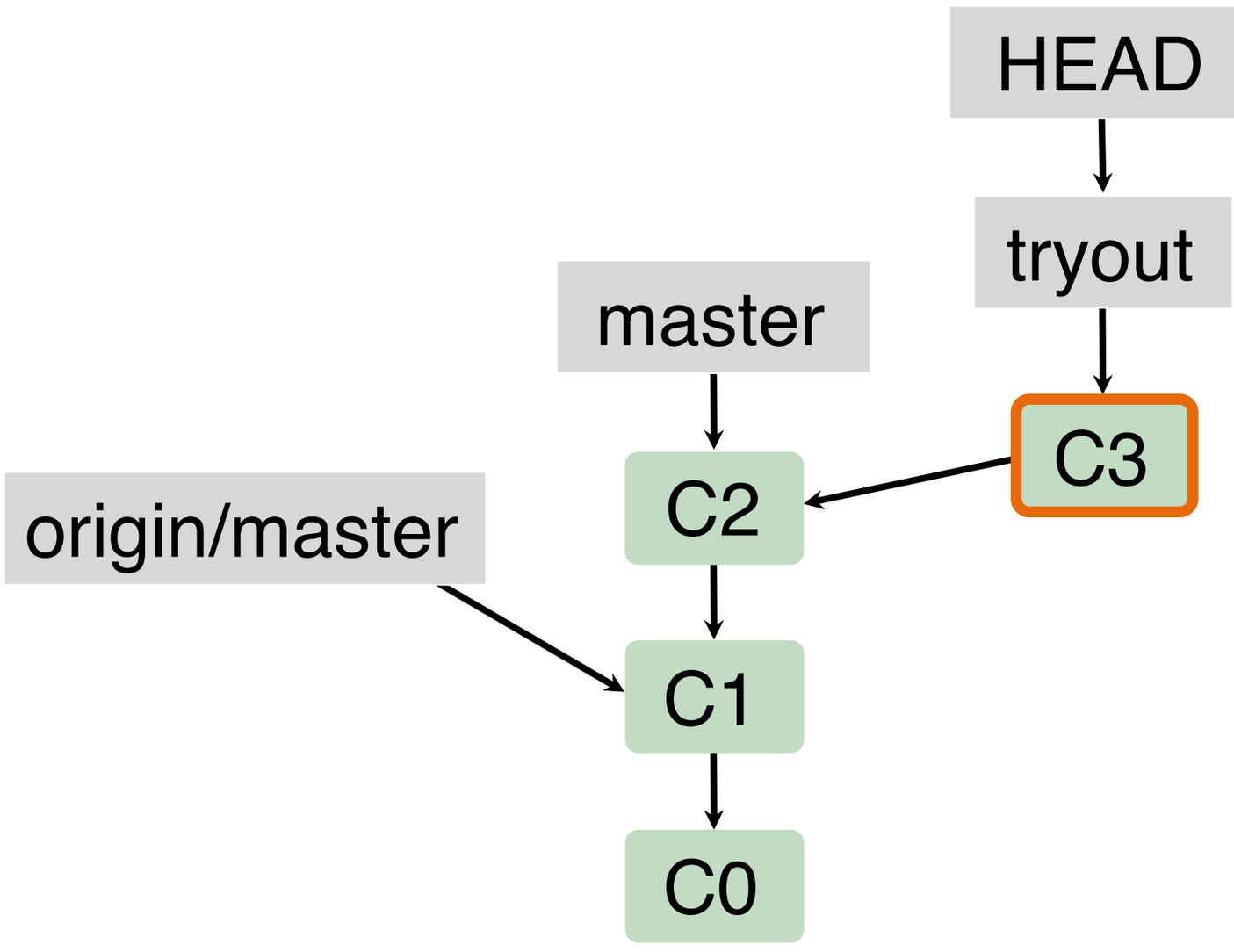
```
git branch tryout
```



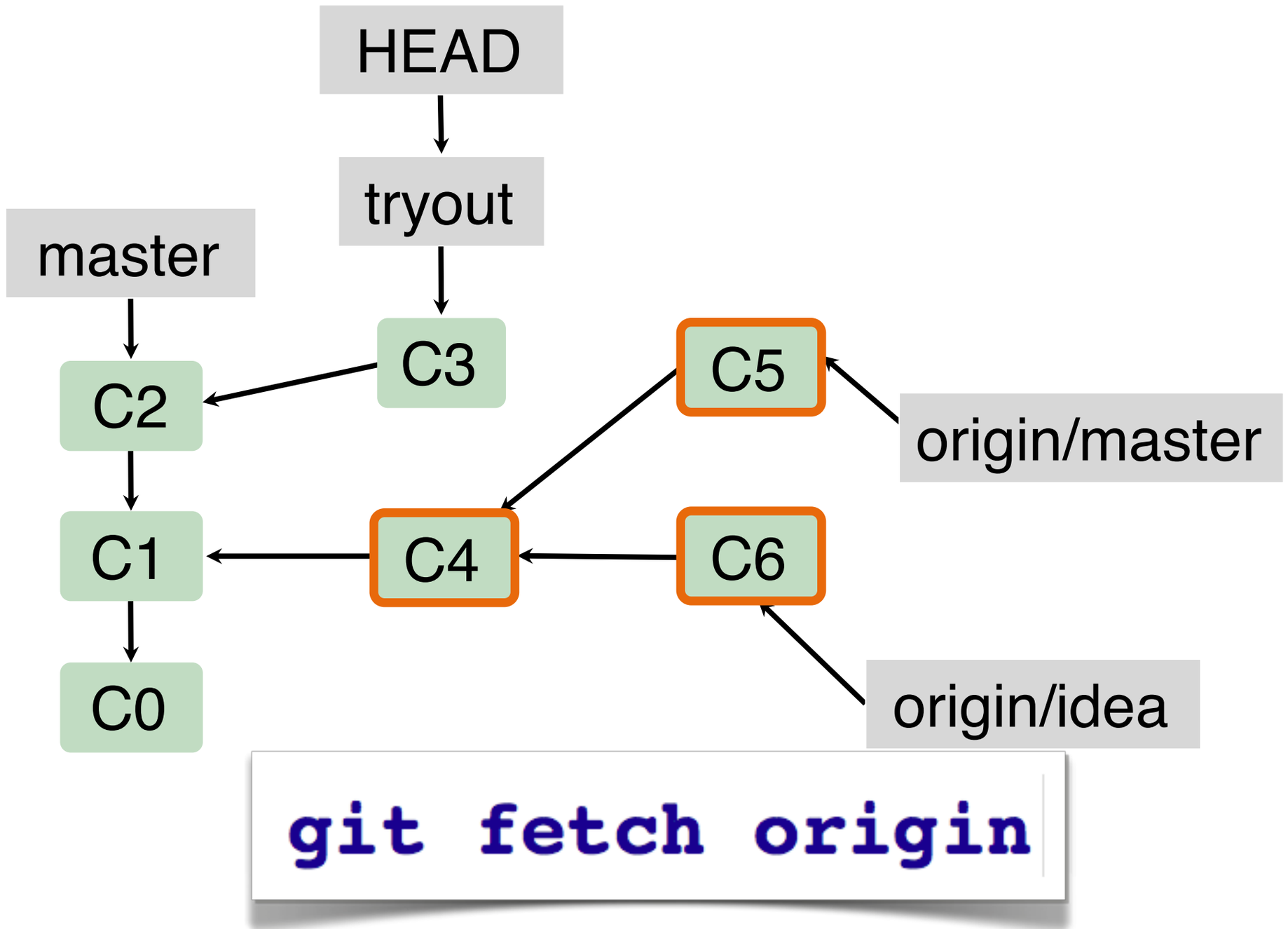
```
git checkout tryout
```



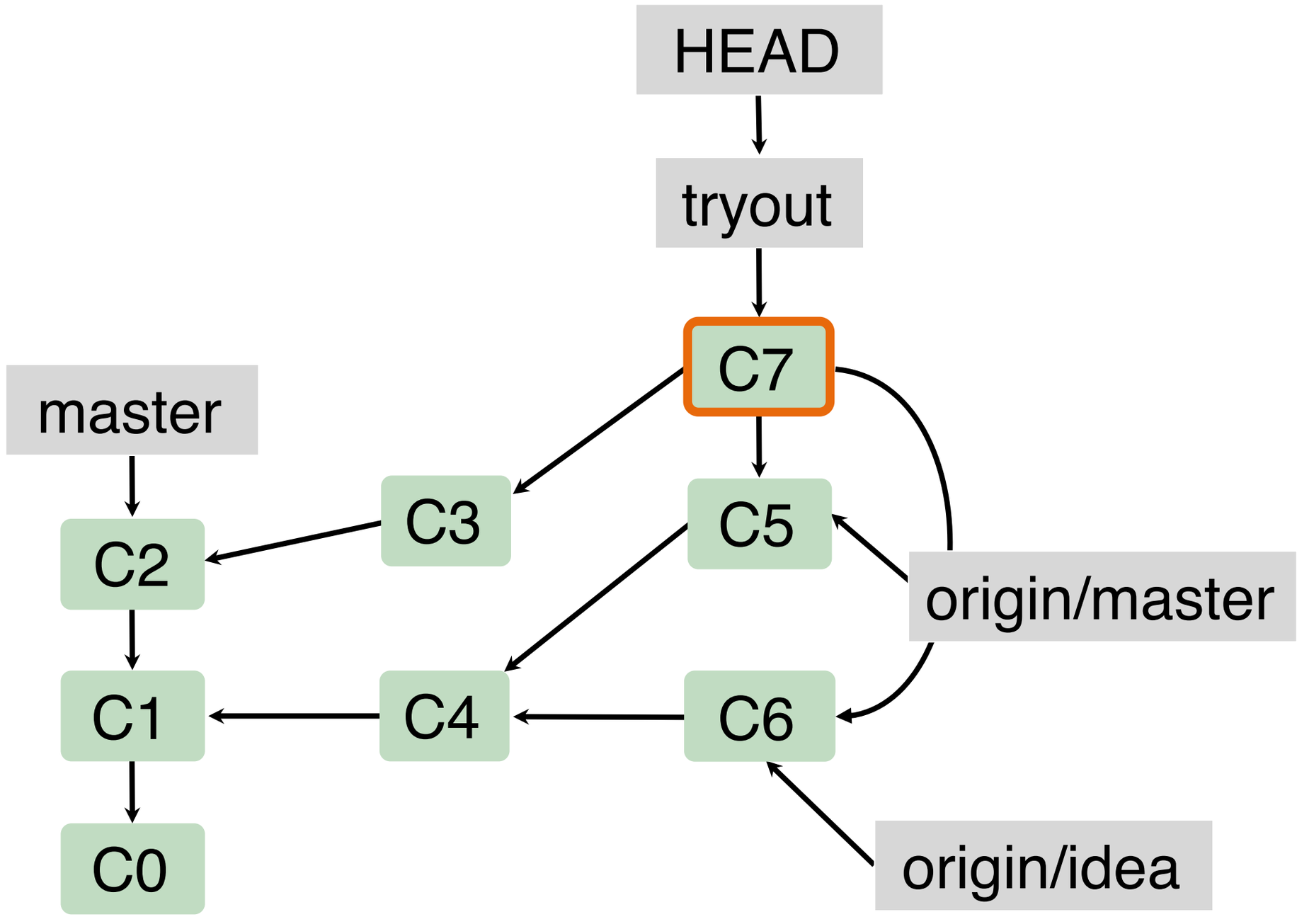
***NB:* git checkout –b … =**
branch + checkout



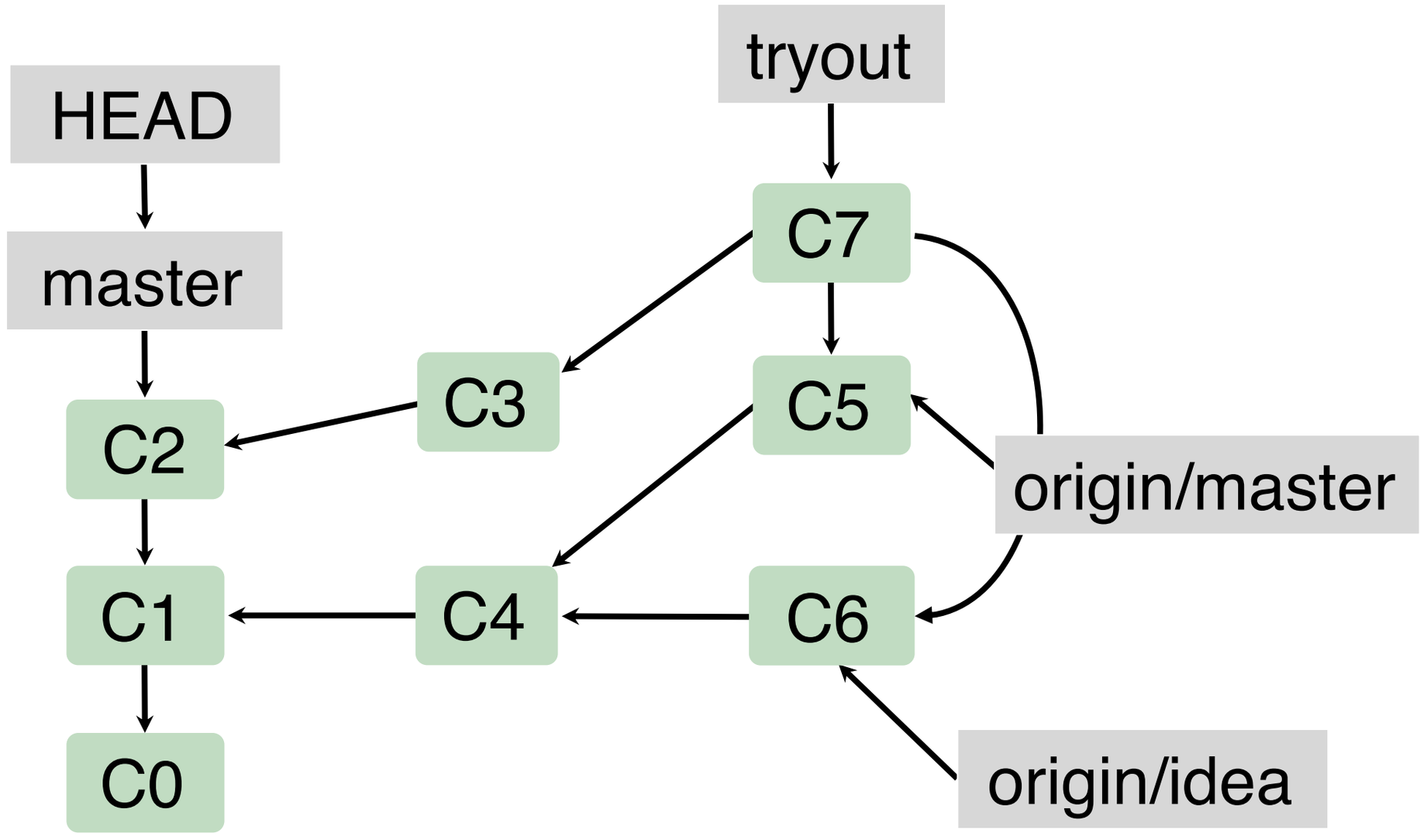

```
git commit …
```


HEAD
tryout
master
C3
C2
C5
origin/master
C1
C4
C6
C0
origin/idea
git fetch origin

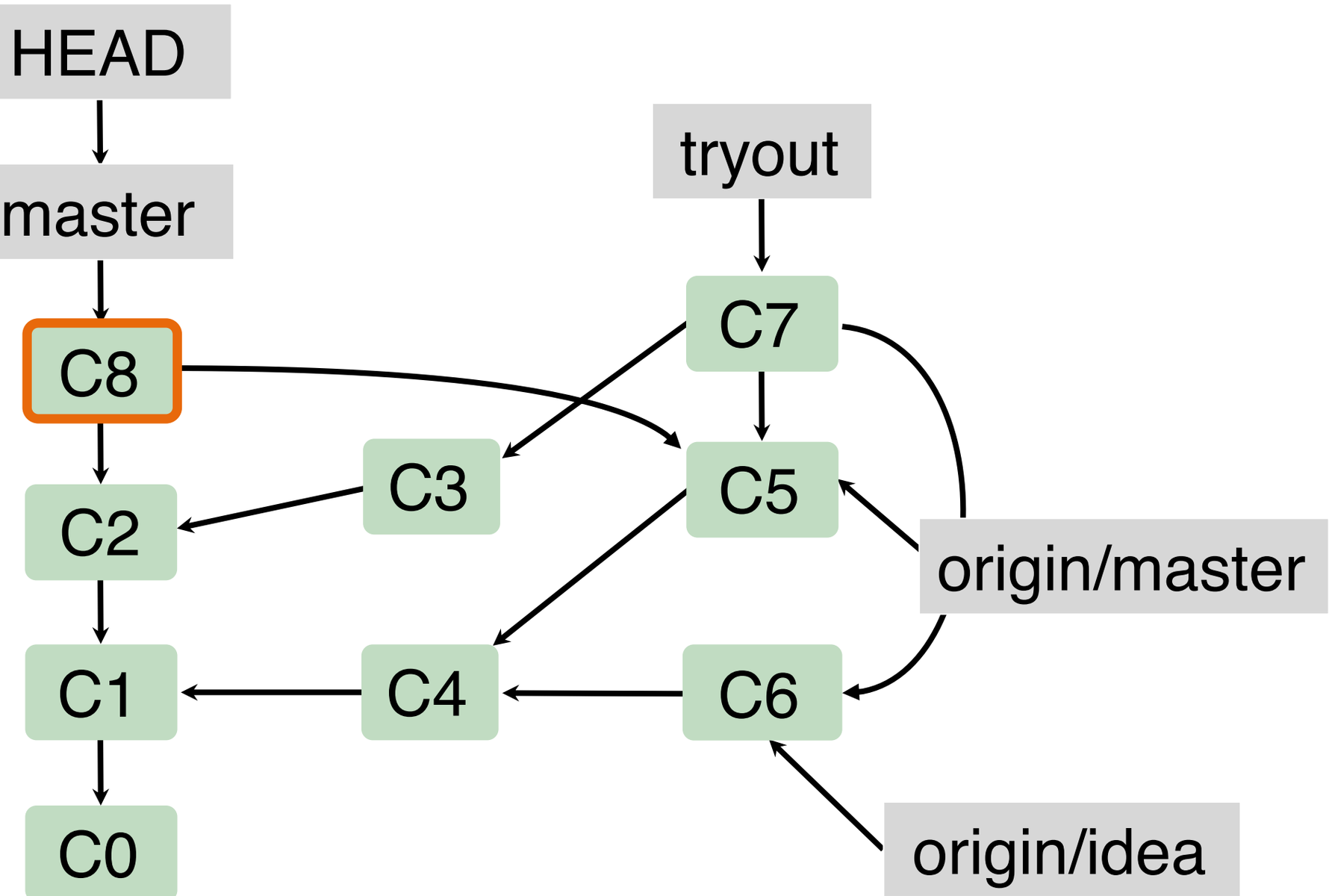
git merge origin/master origin/idea



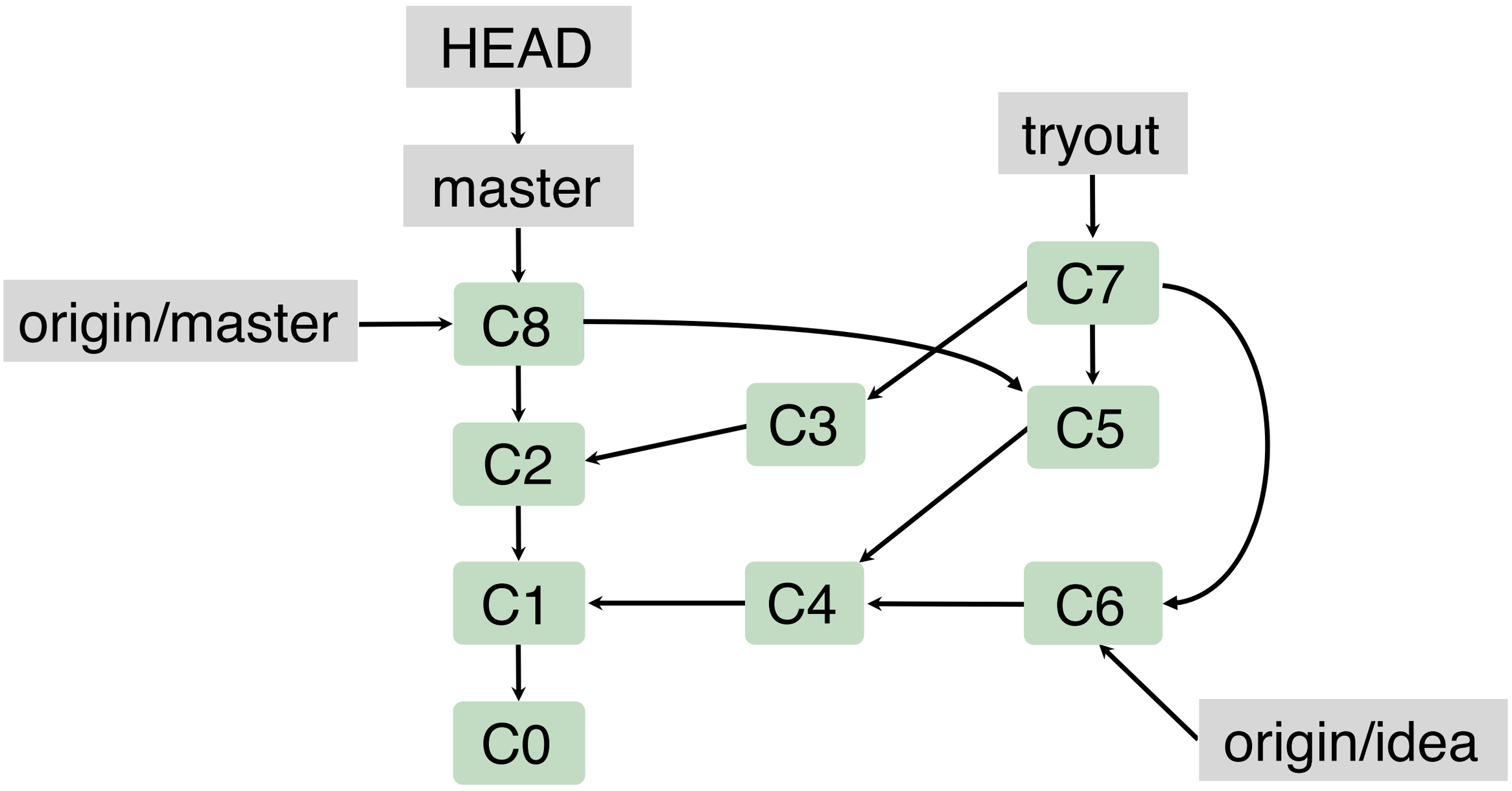
```
git checkout master
```

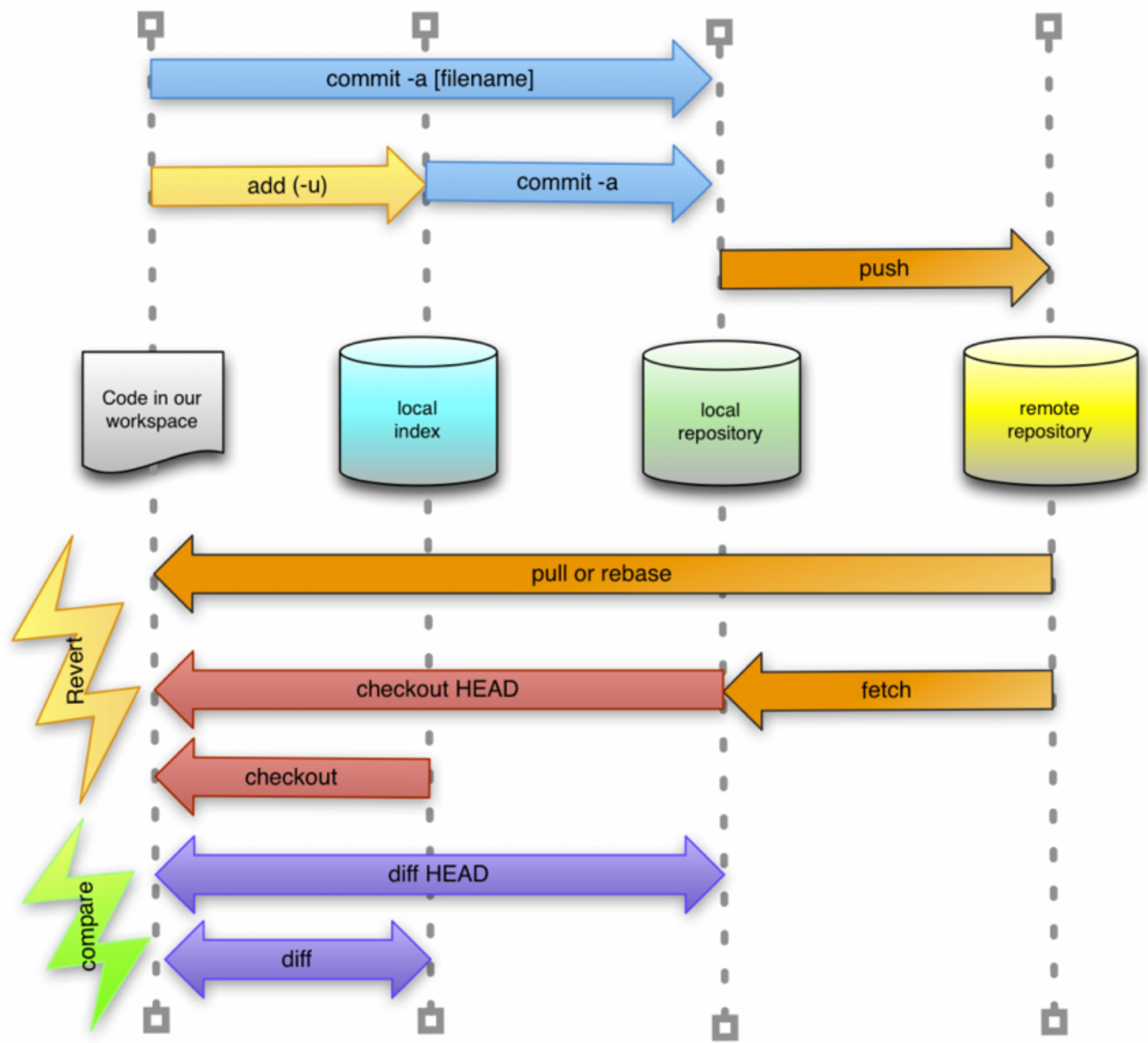


`git merge`


HEAD
master
C8
C2
C1
C0
C3
C4
tryout
C7
C5
C6
origin/master
origin/idea

```
git push
```


HEAD
master
origin/master
C8
C3
C2
C1
C0
tryout
C7
C5
C4
C6
origin/idea


commit -a [filename]
add (-u)
commit -a
push
Code in our workspace
local index
local repository
remote repository
pull or rebase
Revert
checkout HEAD
fetch
checkout
compare
diff HEAD
diff

Homework

- Get a free account on github.com
- Do the exercises on https://github.com/fugufisch/hu_bp_python_course/blob/master/02_introduction/exercise.md