

Git 101

Scott Chacon

Introduction

Me

Scott Chacon



github
SOCIAL CODE HOSTING

git-scm.com

[Home](#)[About Git](#)[Documentation](#)[Download](#)[Related Tools](#)[Wiki](#)

The [Git User's Survey 2008](#) is up! Please devote a few minutes of your time to fill it out, so we can improve Git!

Git is...

Git is an **open source, distributed version control system** designed to handle everything from small to very large projects with speed and efficiency.

Every Git clone is a full-fledged repository with complete history and full revision tracking capabilities, not dependent on network access or a central server.
Branching and merging are fast and easy to do.

Git is used for version control of files, much like tools such as [Mercurial](#), [Subversion](#), [CVS](#), [Perforce](#), [Bitkeeper](#), and [Visual SourceSafe](#).

Projects using Git

- [Git](#)
- [Linux Kernel](#)
- [Ruby on Rails](#)
- [WINE](#)
- [Fedora](#)
- [X.org](#)
- [Rubinius](#)
- [VLC](#)
- [Prototype](#)

Download Git

The latest stable Git release is

v1.6.0.2

[release notes](#) (2008-09-12)



[tar.bz2 \(sign\)](#)



[tar.gz \(sign\)](#)

[Other Download Options](#)

[Source and History](#)

Git Quick Start

Cloning and Creating a Patch

```
$ git clone git://github.com/git/hello-world.git  
$ cd hello-world
```

Creating and Committing

```
$ cd (project-directory)  
$ git init
```

Got a Question?

If you are curious about something, feel free to ask on the [IRC](#) channel or the [mailing list](#).

Git Reference

[Reference](#) [About](#) § [Site Source](#)

Getting and Creating Projects

- [init](#)
- [clone](#)

Basic Snapshotting

- [add](#)
- [status](#)
- [diff](#)
- [commit](#)
- [reset](#)
- [rm, mv](#)

Branching and Merging

- [branch](#)
- [checkout](#)
- [merge](#)
- [log](#)
- [tag](#)

Sharing and Updating Projects

- [fetch, pull](#)

INTRODUCTION TO THE GIT REFERENCE

This is the Git reference site. This is meant to be a quick reference for learning Git commands. The commands are organized into sections of the type of options and commands needed to accomplish these common tasks.

Each section will link to the next section, so it can be used as a tutorial. Even as the official manual pages and relevant sections in the **Pro Git book**, so with thinking about source code management like Git does.

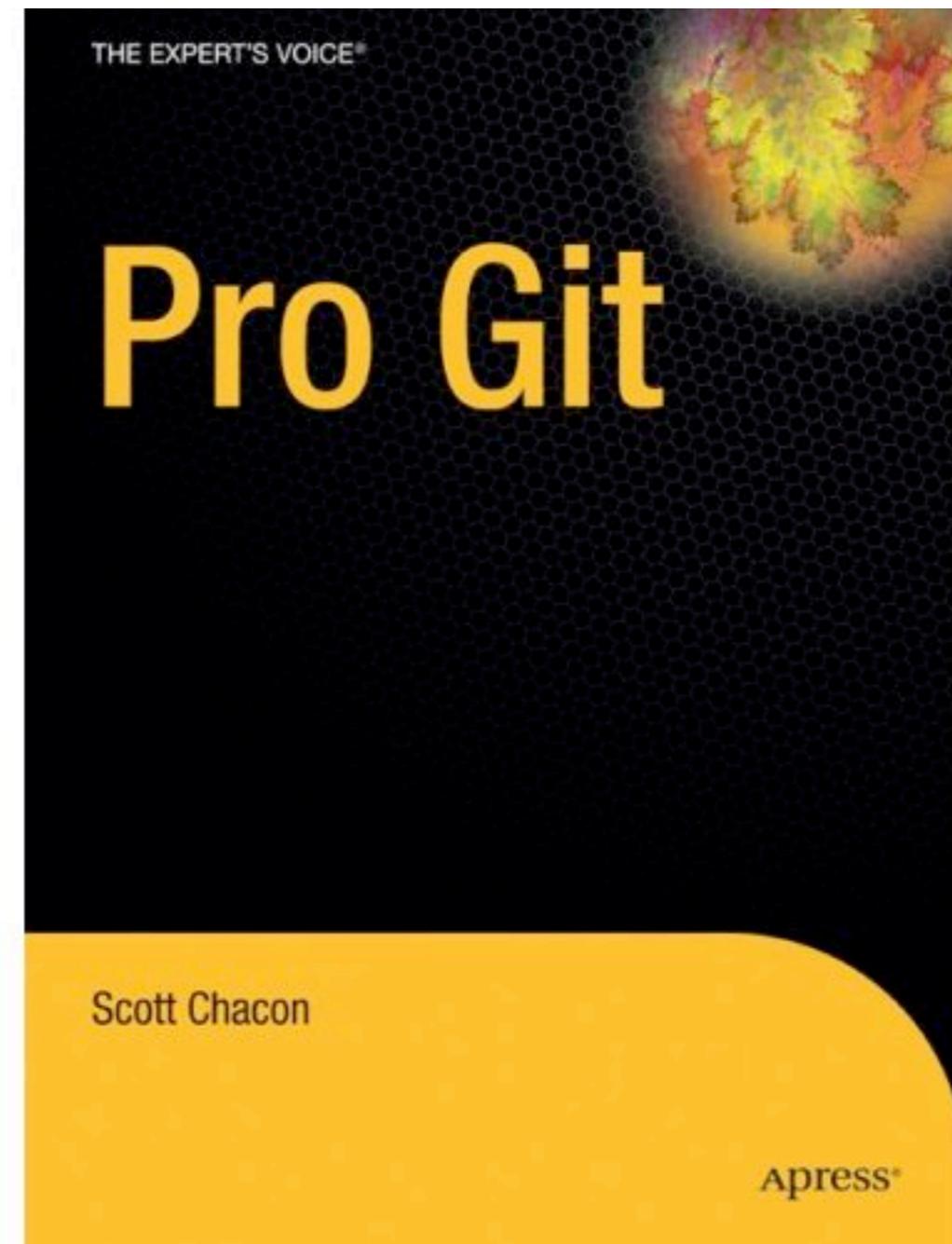
HOW TO THINK LIKE GIT

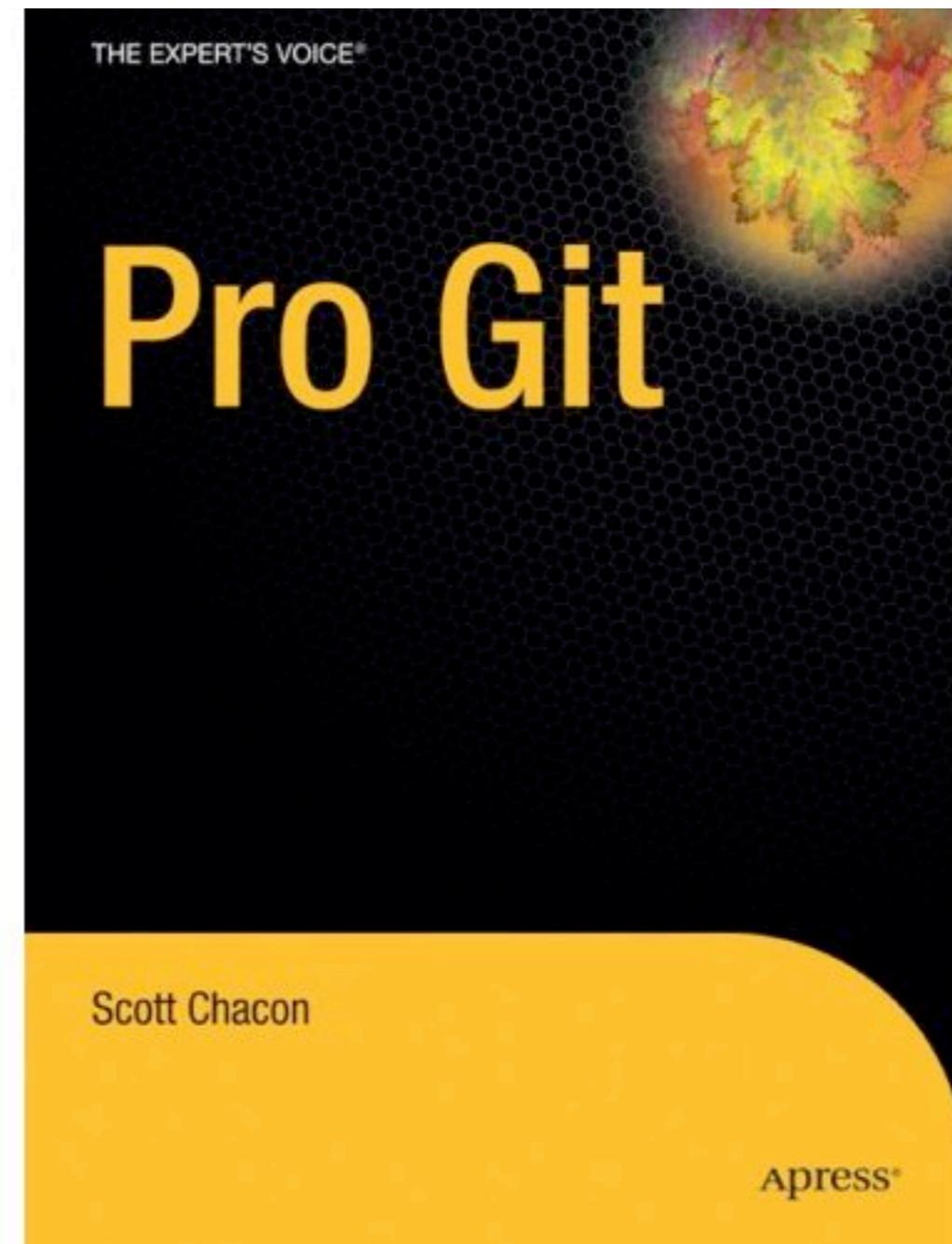
This first thing that is important to understand about Git is that it thinks about whatever SCM you may be used to. It is often easier to learn Git by trying to think about it in the Git way.

Let's start from scratch. Assume you are designing a new source code manager. What would you do? Chances are that you simply copied your project directory.

```
$ cp -R project project.bak
```

<http://gitref.org>





<http://progit.org>

schacon@gmail.com

@chacon

twitter

</me>

What is Git?

“distributed version
control system”

what is version control?

what is version control?

what do we use it for?

**imagine a world with no
version control tools**



**WON'T SOMEBODY PLEASE
THINK OF THE CHILDREN!?**

what would you do?

downsides?

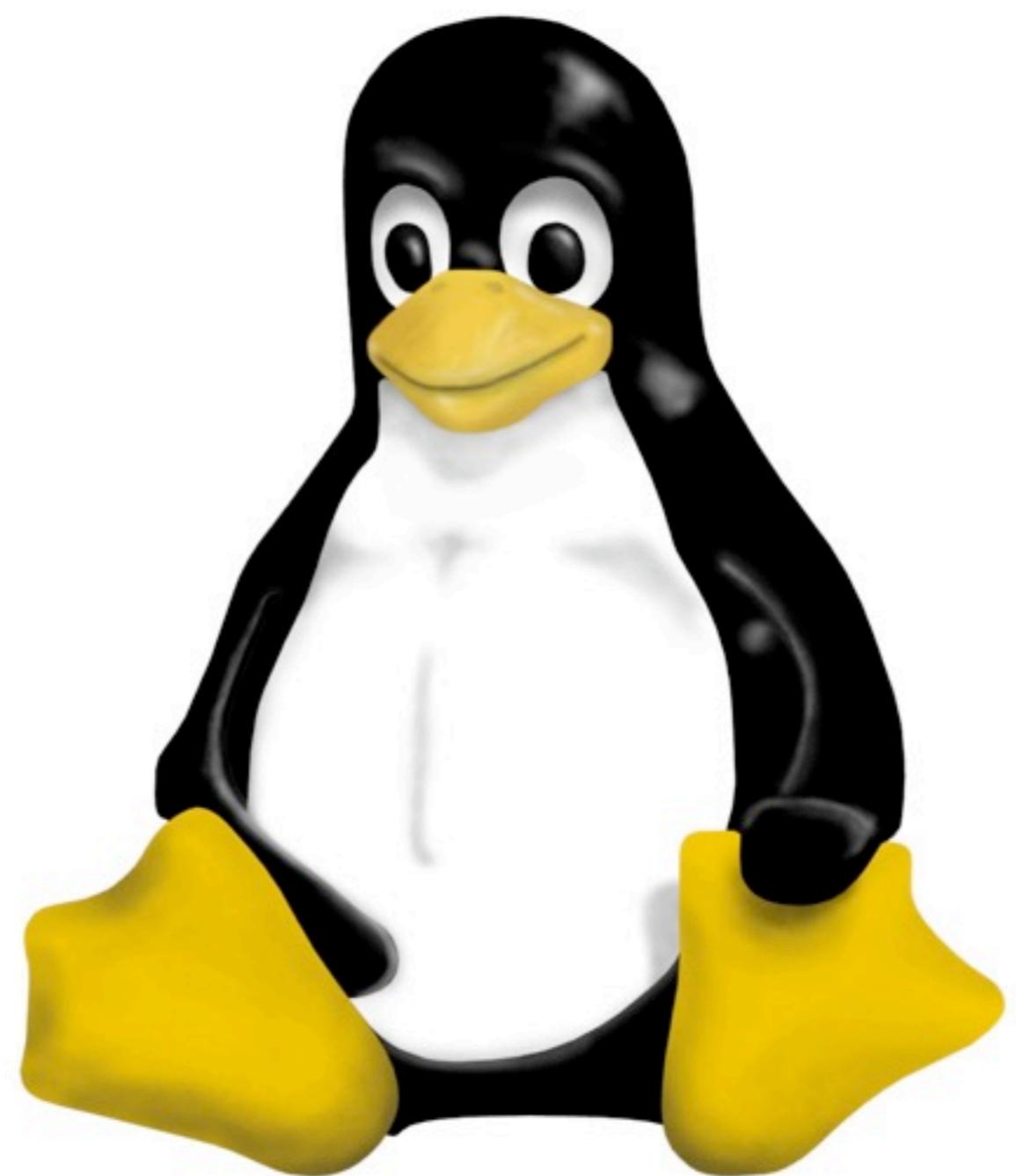
who doesn't use version control?

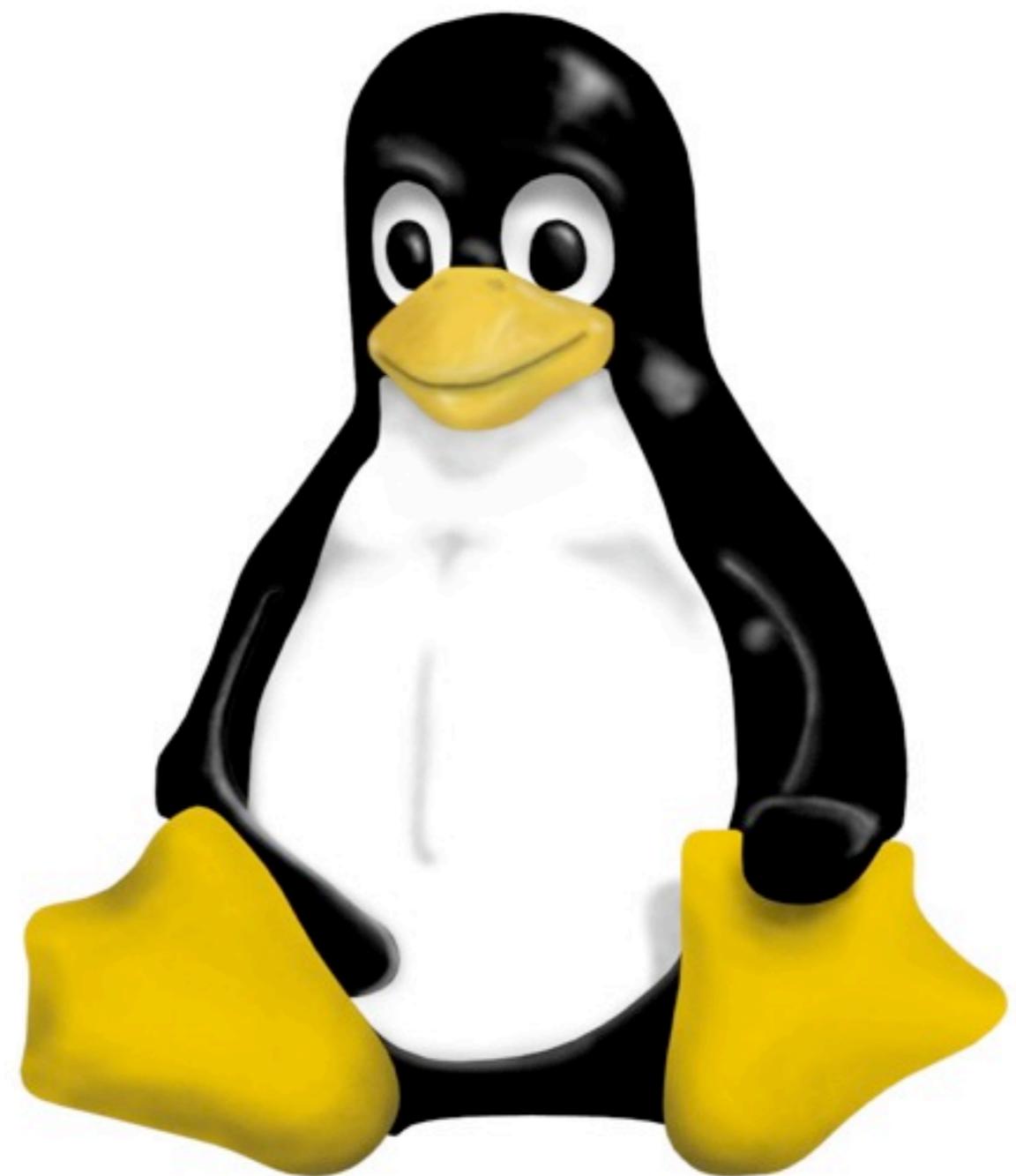
who doesn't use version control?

rhetorical question

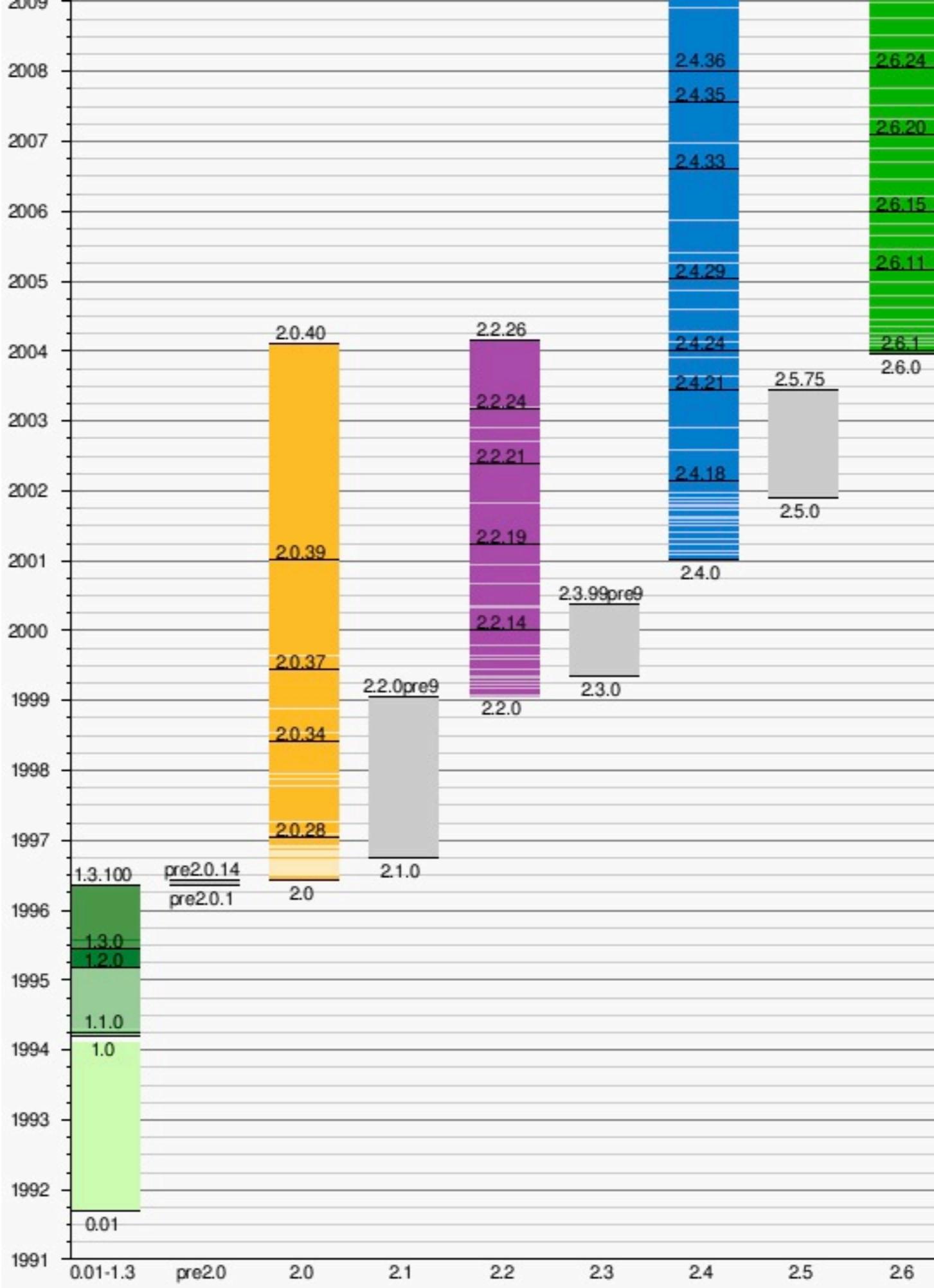
what project is this?

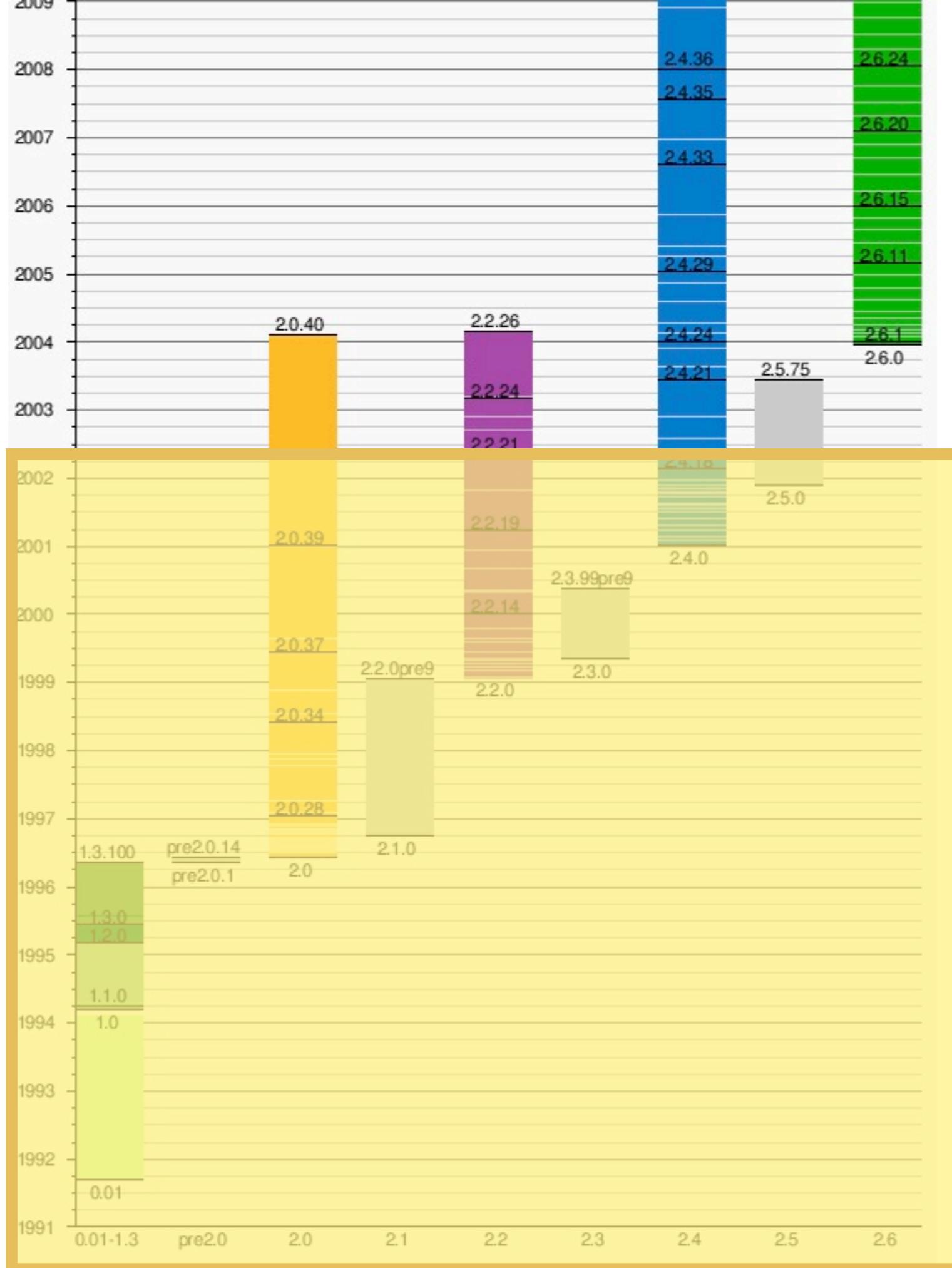
- thousands of developers
- working all across the world
- used on millions of computers
- runs 90% of the supercomputers in the world
- runs my phone
- **I I years with no version control system**
- has an adorable mascot

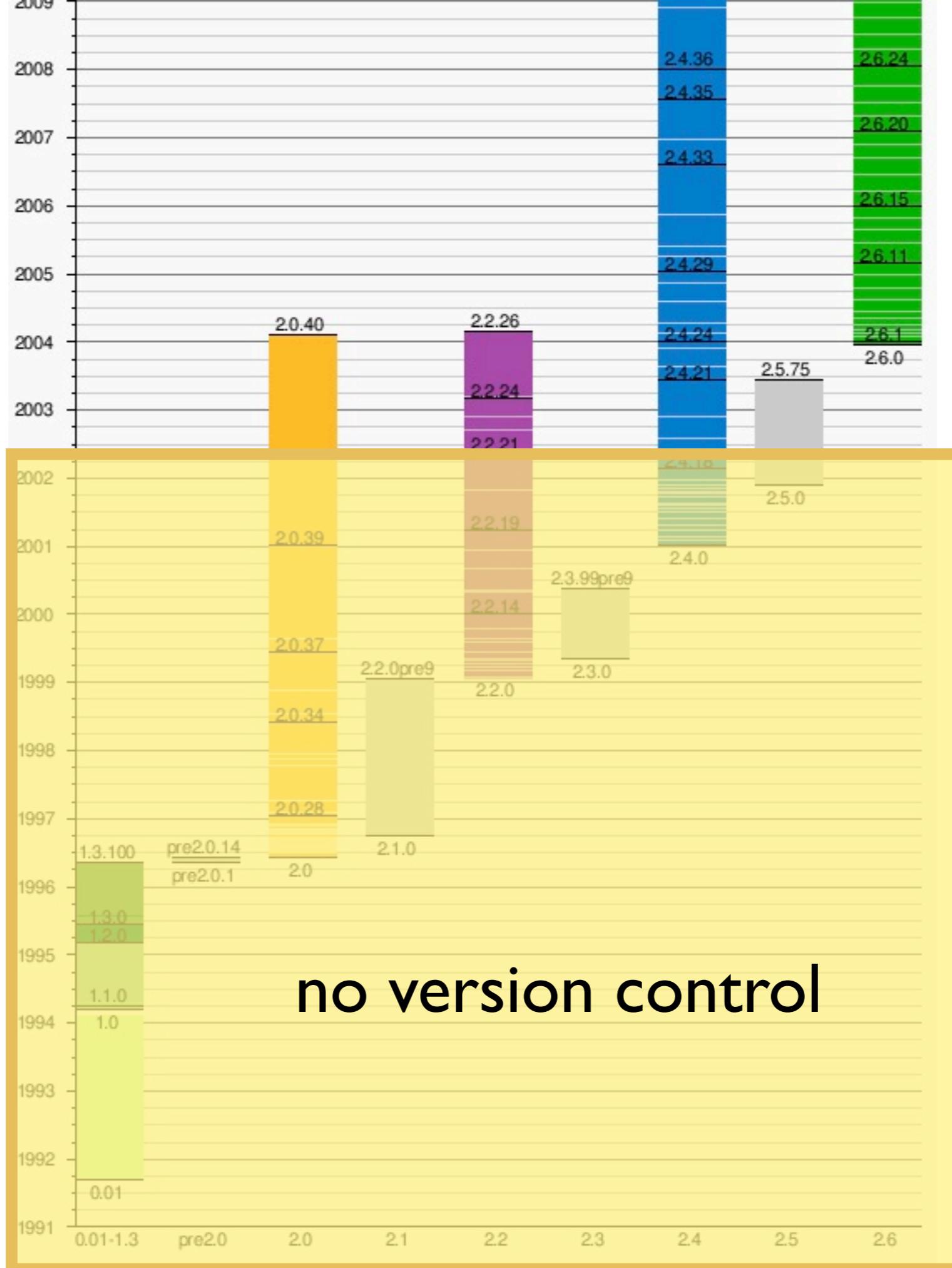




| 1991-2002







how did they do it?

main dude

main.c

library.c

library.h

main dude

main.c

library.c

library.h



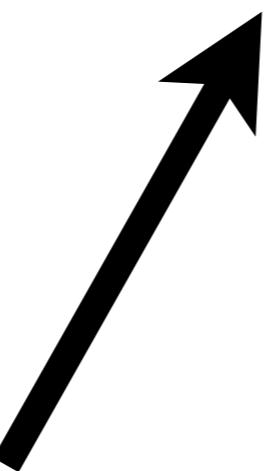
ver-1.0.zip

main dude

you

main.c
library.c
library.h

ver-1.0.zip



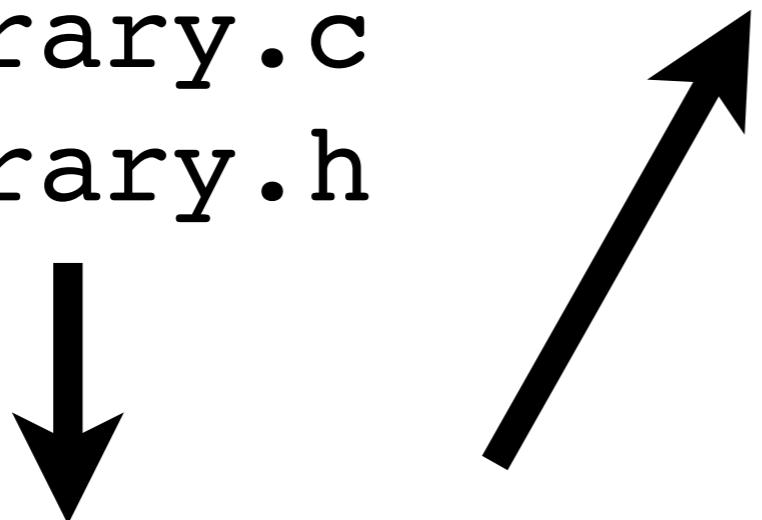
main dude

you



main.c
library.c
library.h

ver-1.0.zip



main dude

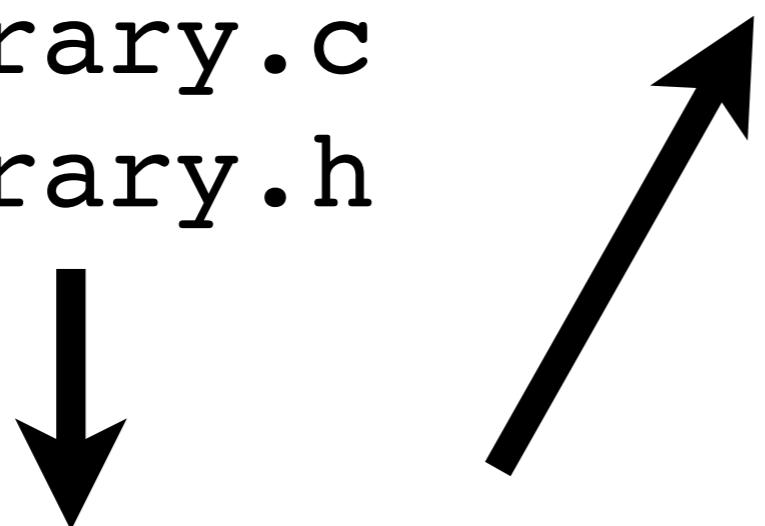
you



main.c
library.c
library.h

v1/main.c
v1/library.c
v1/library.h

ver-1.0.zip



main dude

you



main.c
library.c
library.h

v1/main.c
v1/library.c
v1/library.h

ver-1.0.zip

cp -Rf v1 patch

main dude

you

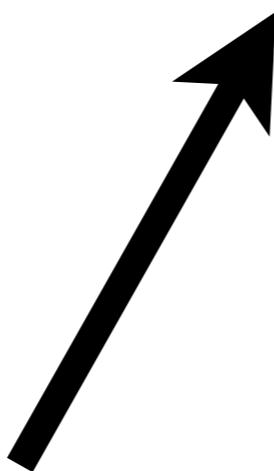


main.c
library.c
library.h



ver-1.0.zip

v1/main.c
v1/library.c
v1/library.h



cp -Rf v1 patch

patch/main.c
patch/library.c
patch/library.h

main dude

you

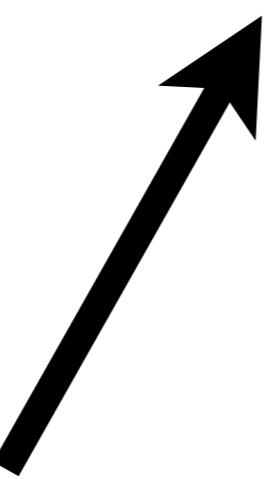


main.c
library.c
library.h



ver-1.0.zip

v1/main.c
v1/library.c
v1/library.h



cp -Rf v1 patch

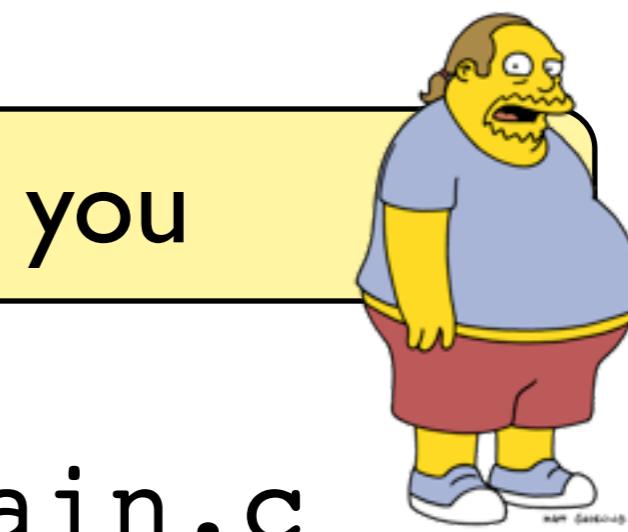
patch/main.c
patch/library.c
patch/library.h

main dude

main.c
library.c
library.h

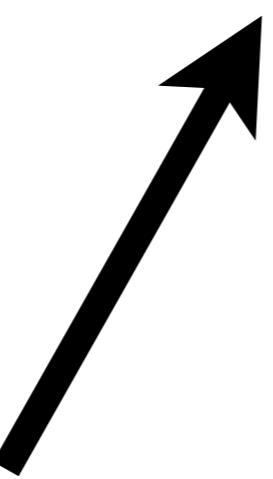


ver-1.0.zip



you

v1/main.c
v1/library.c
v1/library.h



cp -Rf v1 patch

patch/main.c
patch/library.c*
patch/library.h

diff -u v1 patch

main dude

you



```
$ git diff v1 patch
diff --git a/v1/library.c b/patch/library.c
index 0be4b5f..0120580 100644
--- a/v1/library.c
+++ b/patch/library.c
@@ -19,7 +19,7 @@ static struct {
    { "detachedhead", &advice_detached_head },
};

-int git_default_config(const char *var, const char *value)
+int YO_MOMMA_IS_FAT(const char *var, const char *value)
{
    const char *k = skip_prefix(var, "advice.");
    int i;
```

diff -u v1 patch

main dude

main.c
library.c
library.h

ver-1.0.zip

you



v1/main.c
v1/library.c
v1/library.h

cp -Rf v1 patch

patch/main.c
patch/library.c*
patch/library.h

email

diff -u v1 patch

main dude

you



main.c
library.c
library.h

v1/main.c
v1/library.c
v1/library.h

ver-1.0.zip

cp -Rf v1 patch

ver-1.1.zip

patch/main.c
patch/library.c*
patch/library.h

diff -u v1 patch

ver-1.0.zip

ver-1.1.zip

ver-1.2.zip

now imagine...

main dude

VI main.c
library.c
library.h

main dude

VI main.c
library.c
library.h

you

VI main.c
library.c
library.h



main dude

VI main.c
library.c
library.h

you

VI main.c
library.c
library.h

main.c
V2 library.c*
library.h



main dude

VI main.c
library.c
library.h

main.c*

V3 library.c
library.h

you

VI main.c
library.c
library.h

main.c
library.c*
library.h



main dude

VI main.c
library.c
library.h

V3 main.c*
library.c
library.h

V2 main.c
library.c*
library.h

you

VI main.c
library.c
library.h

V2 main.c
library.c*
library.h



main dude

VI main.c
library.c
library.h

V3 main.c*
library.c
library.h

V4 main.c*
library.c*
library.h

you

VI main.c
library.c
library.h

V2 main.c
library.c*
library.h



main dude

VI main.c
library.c
library.h

V3 main.c*
library.c
library.h

V4 main.c*
library.c*
library.h

you

VI main.c
library.c
library.h

V2 main.c
library.c*
library.h



main dude

VI main.c
library.c
library.h

V3 main.c*
library.c
library.h

V4 main.c*
library.c*
library.h

you

VI main.c
library.c
library.h

V2 main.c
library.c*
library.h



main dude

VI main.c
library.c
library.h

V3 main.c*
library.c
library.h

V2 main.c
library.c*
library.h

V4 main.c*
library.c*
library.h

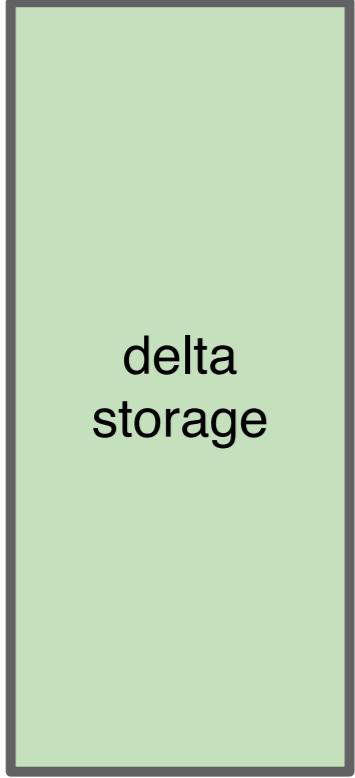
you

VI main.c
library.c
library.h

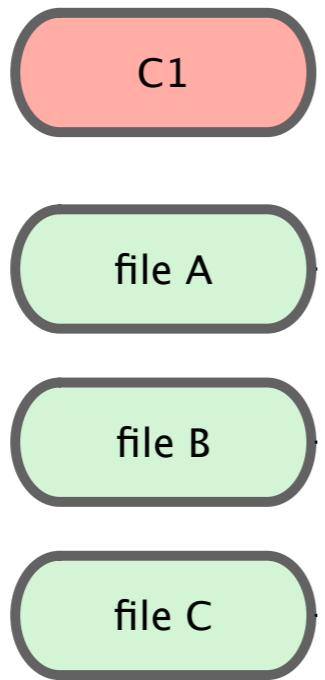
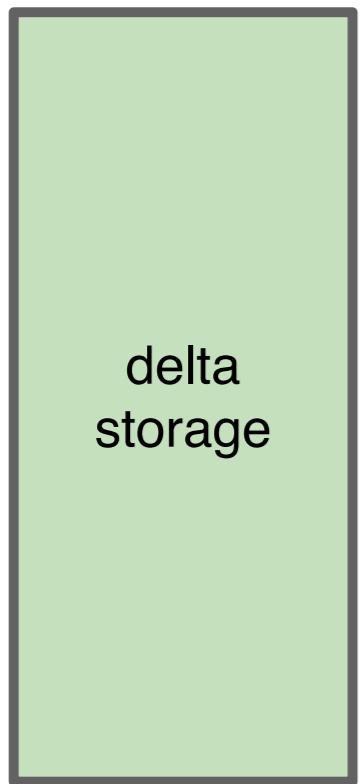
V2 main.c
library.c*
library.h

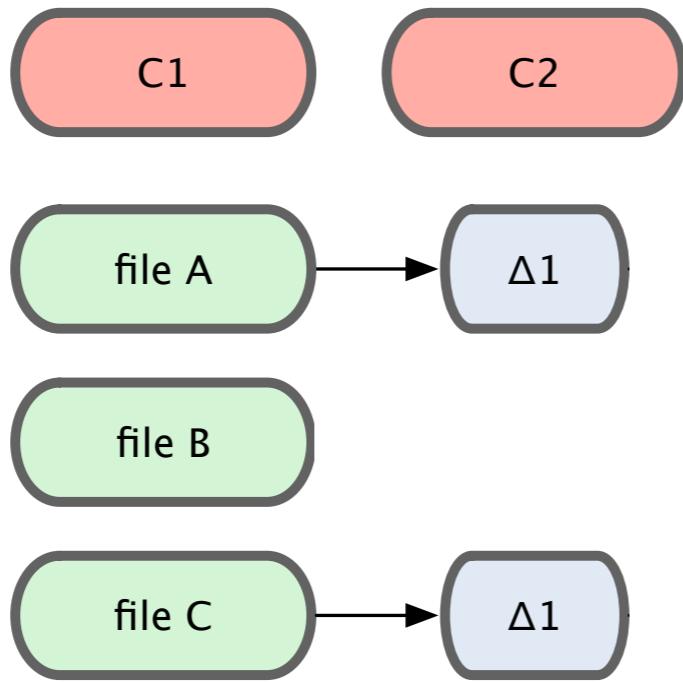
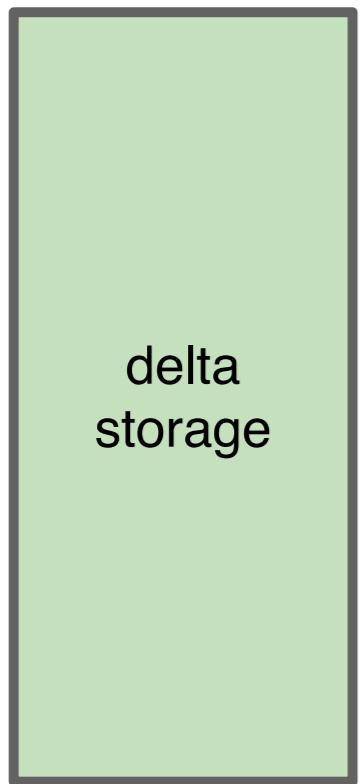


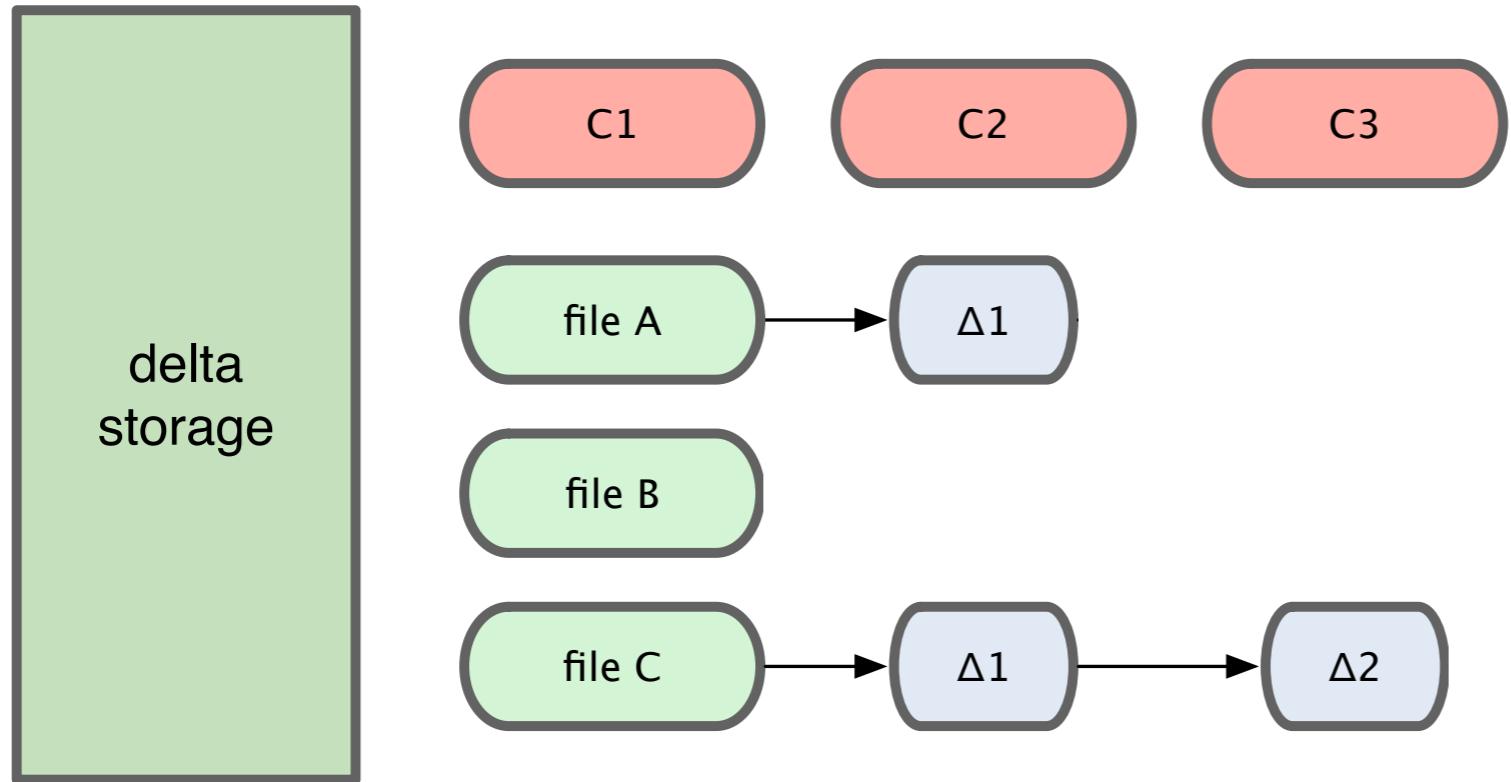
Snapshots, not Patches

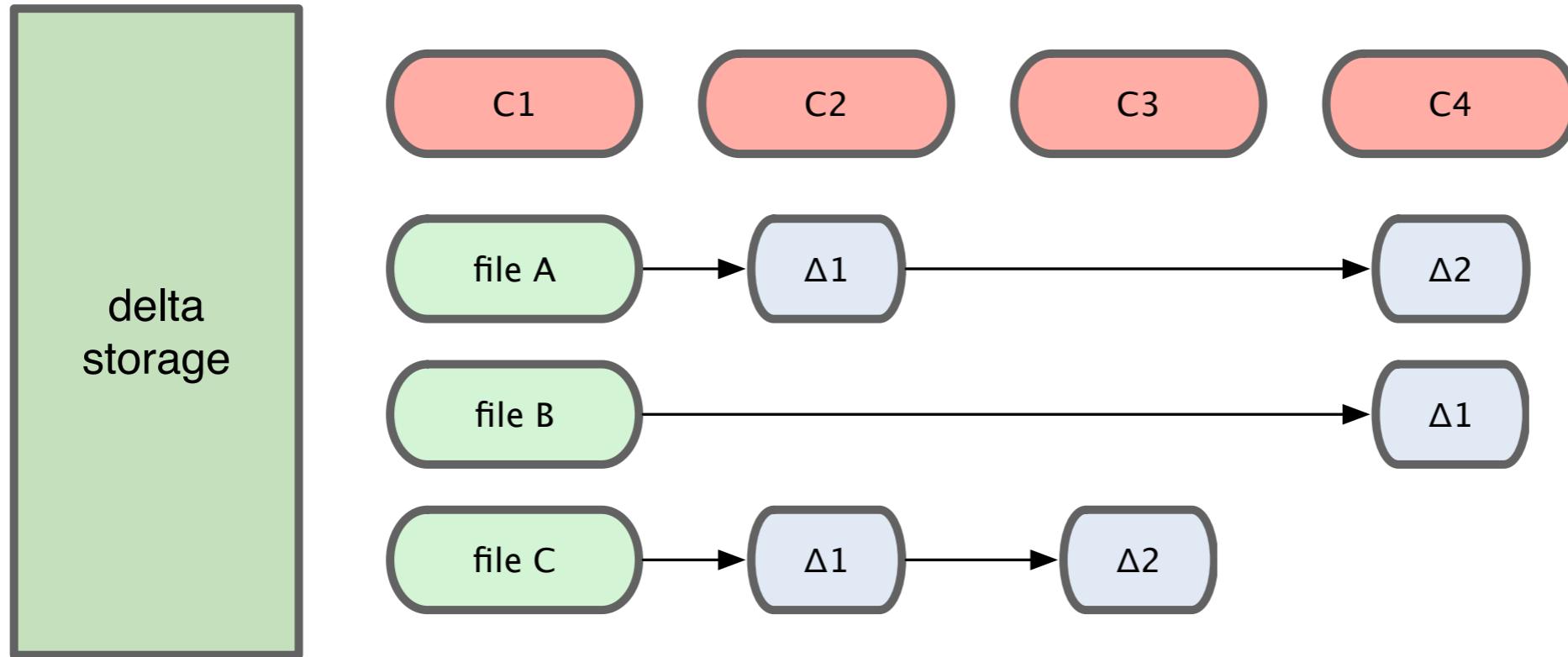


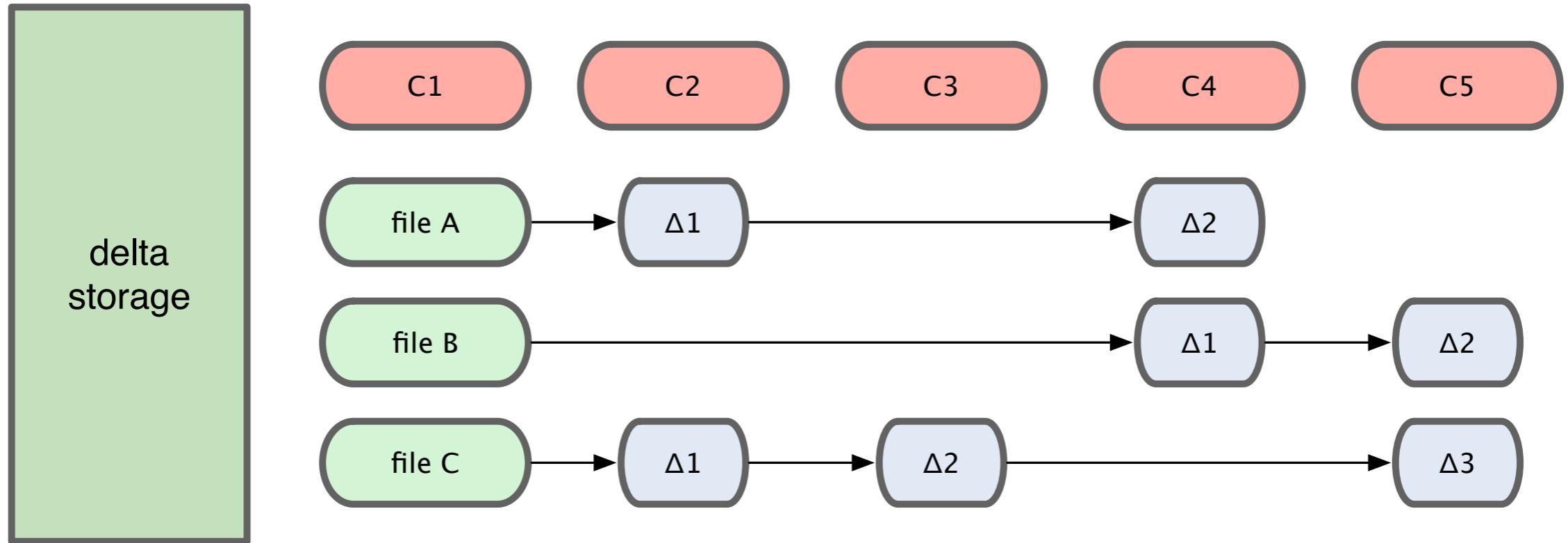
delta
storage

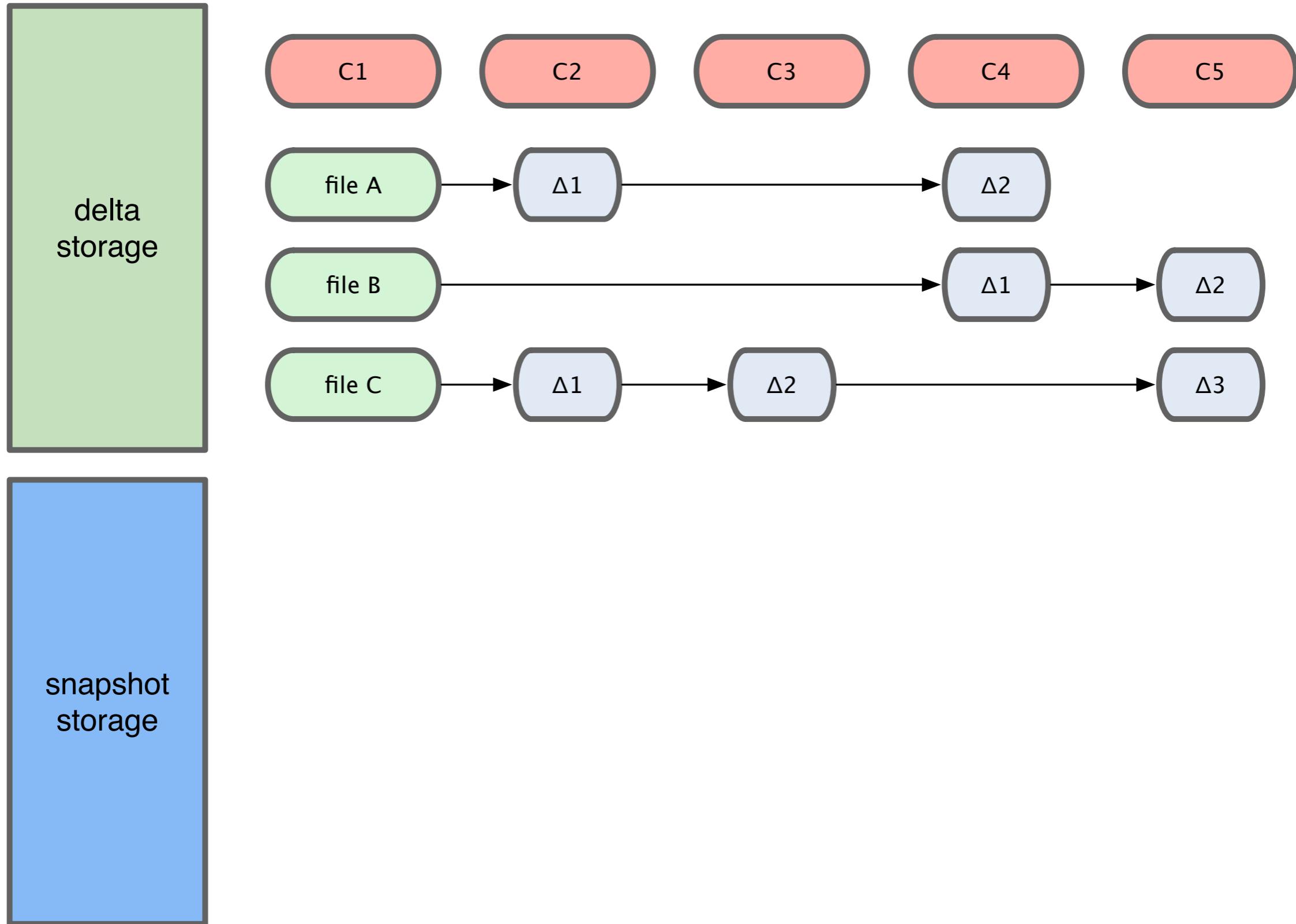


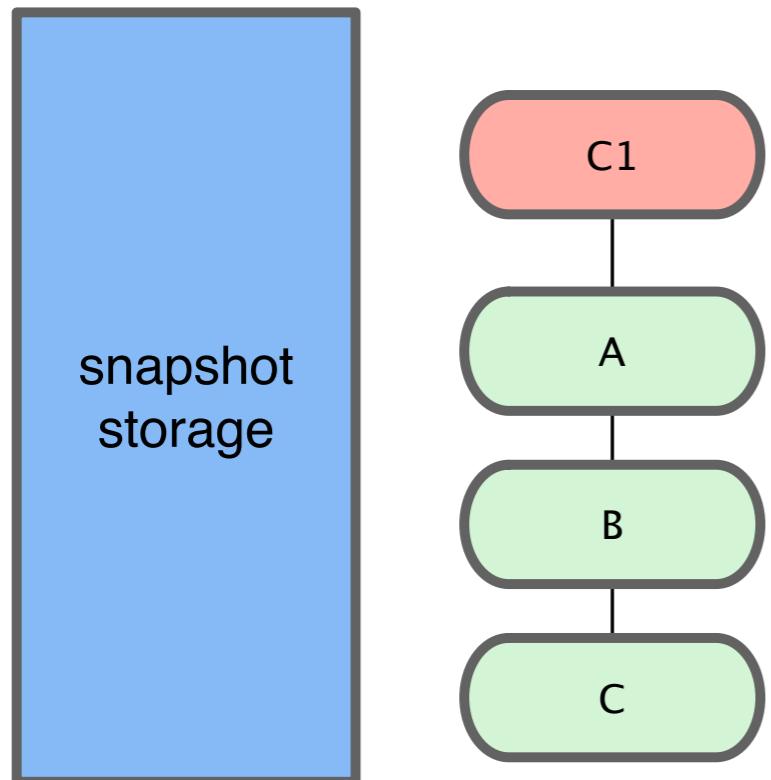
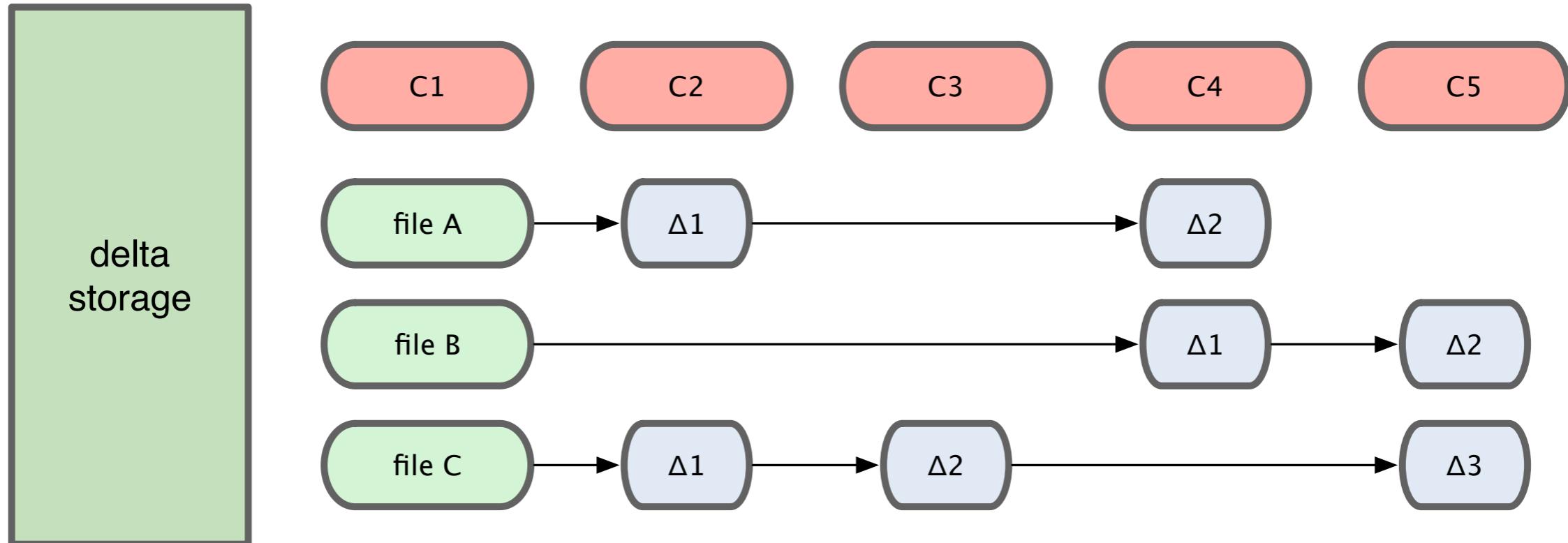


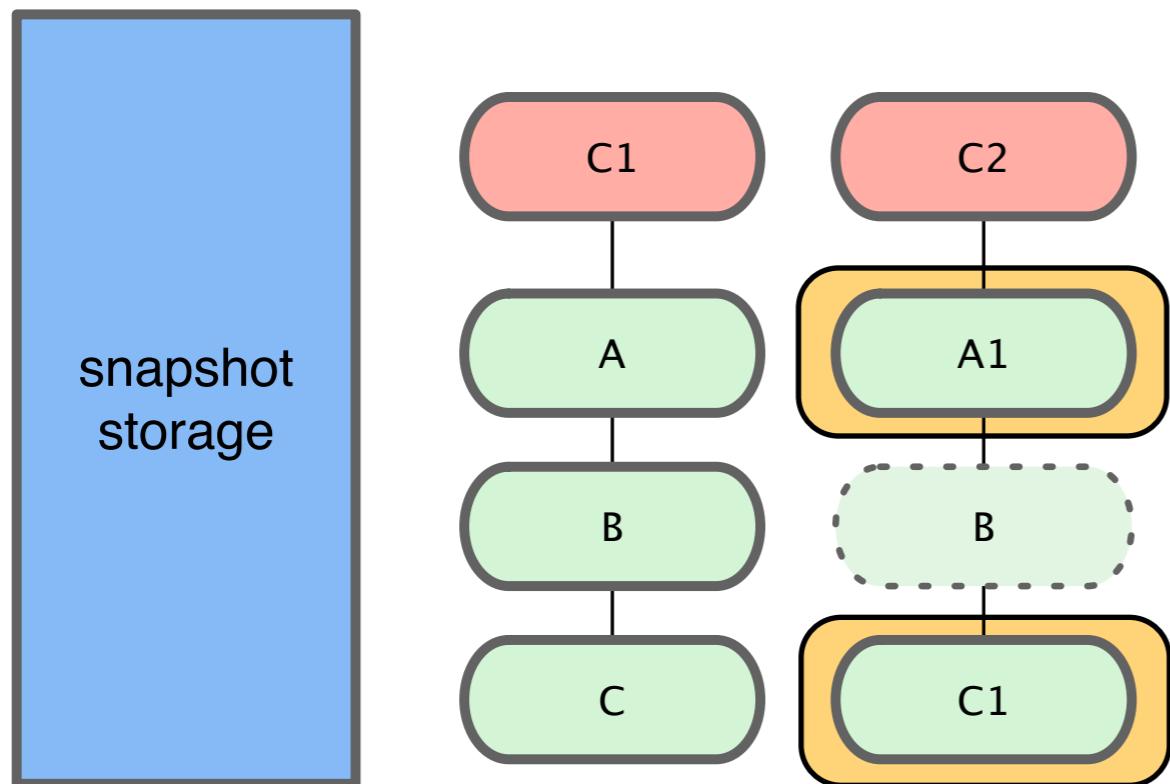
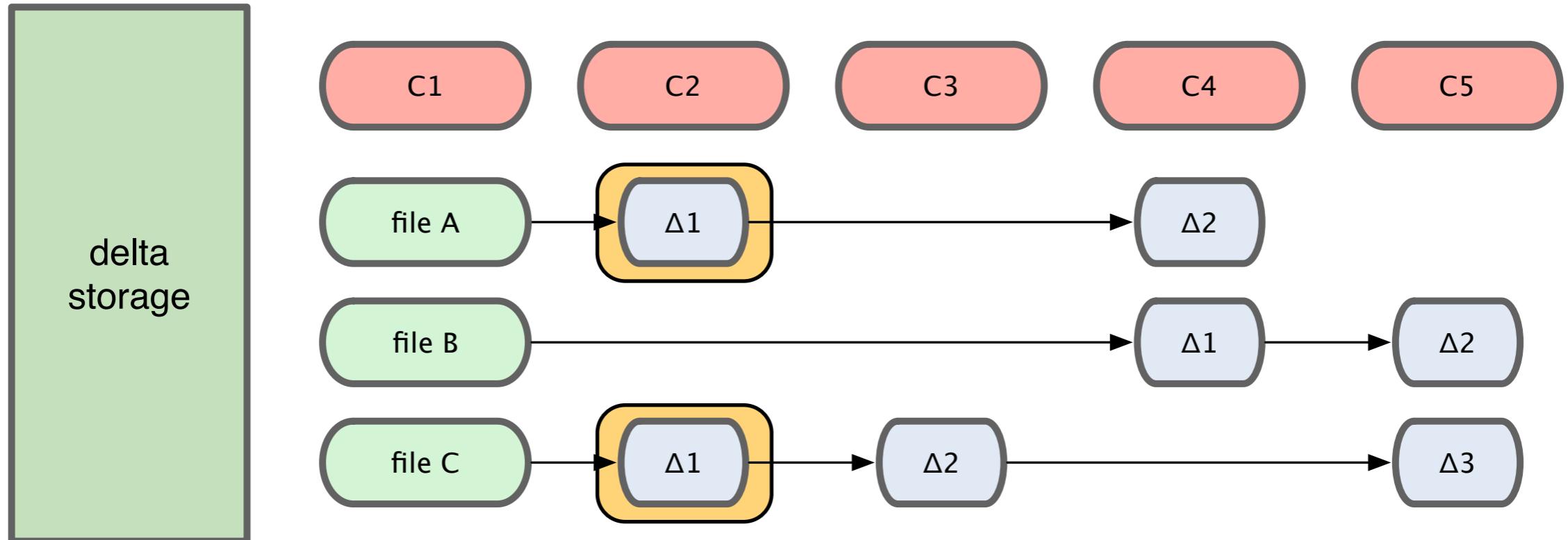


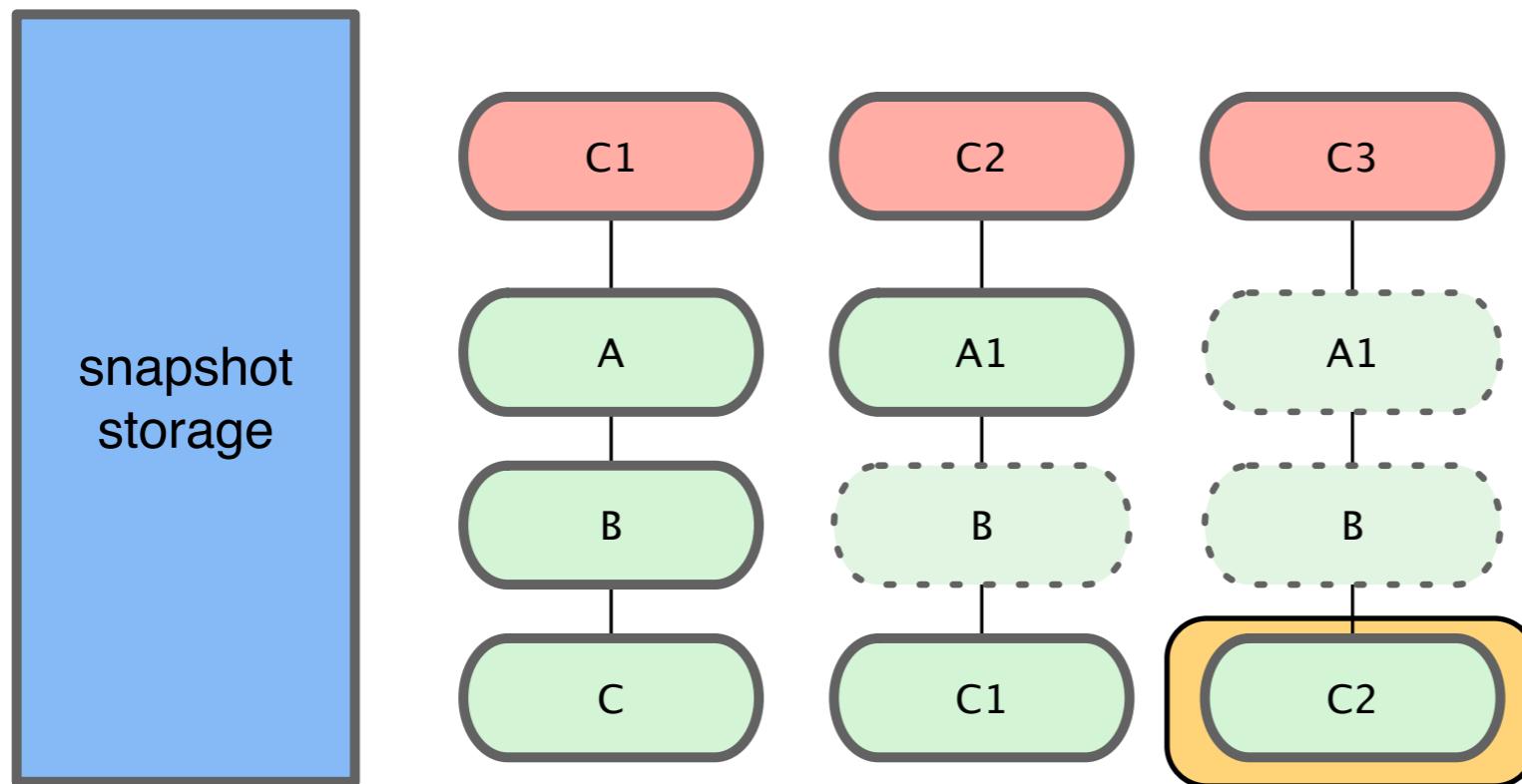
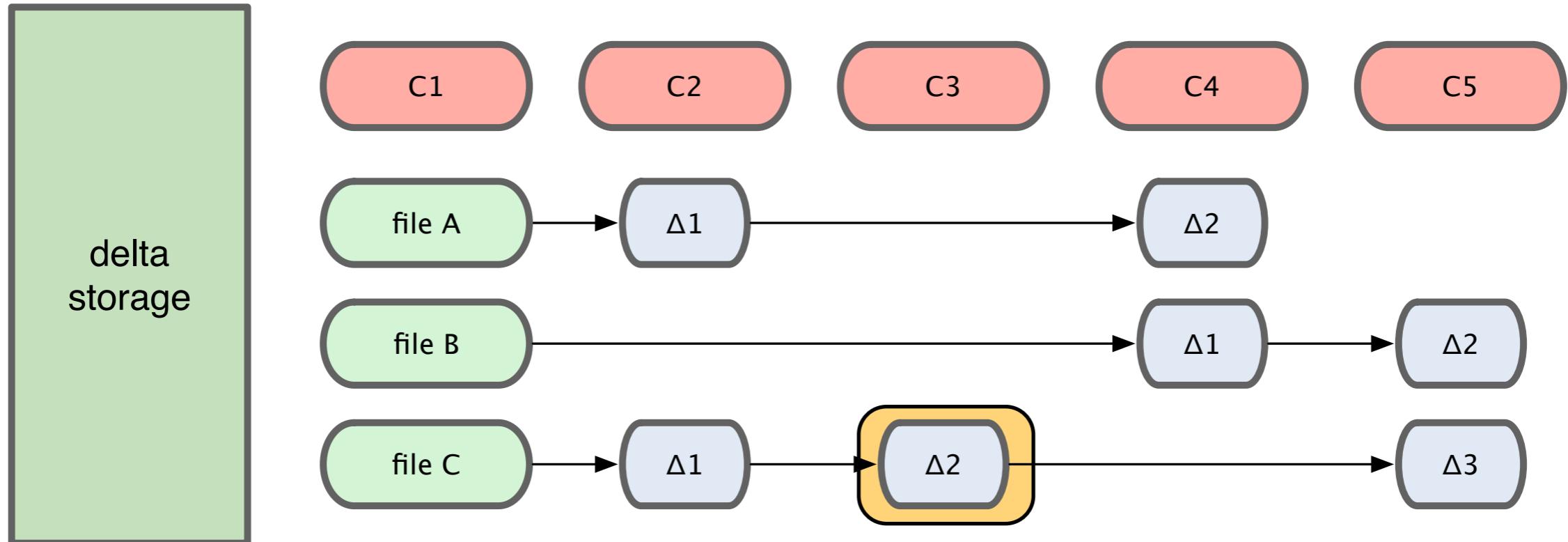


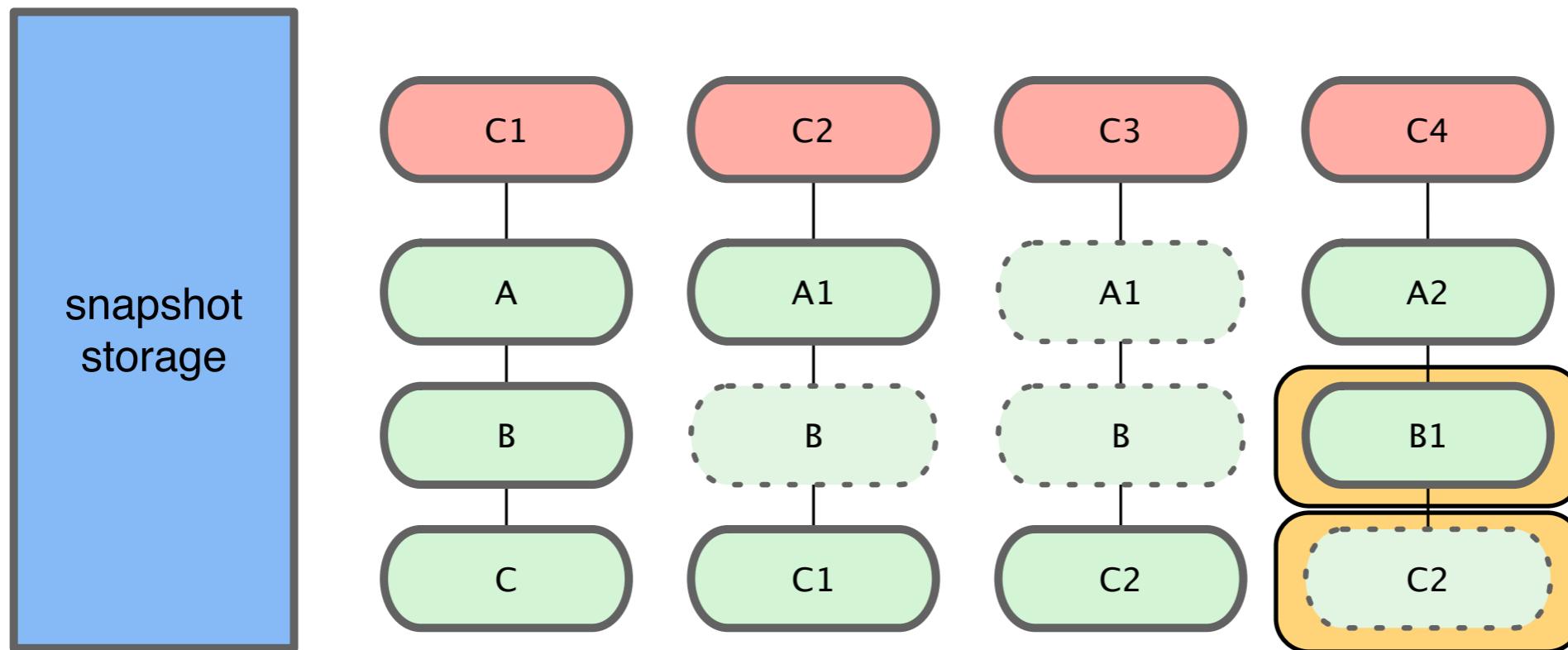
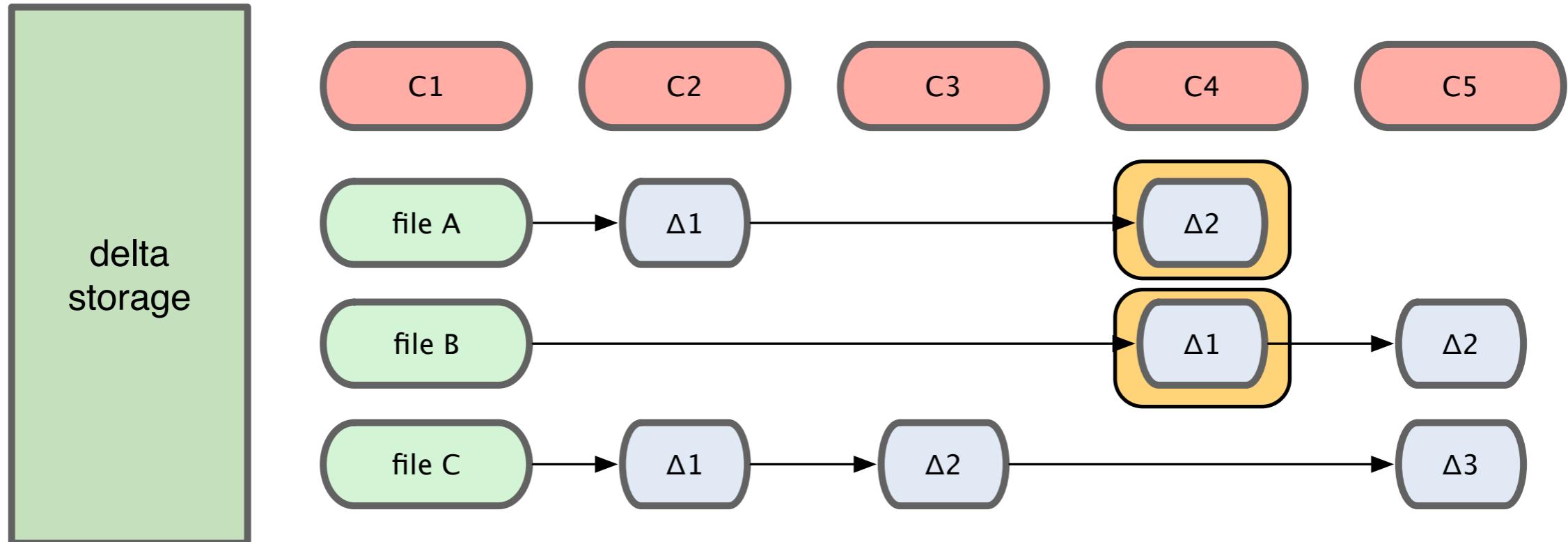


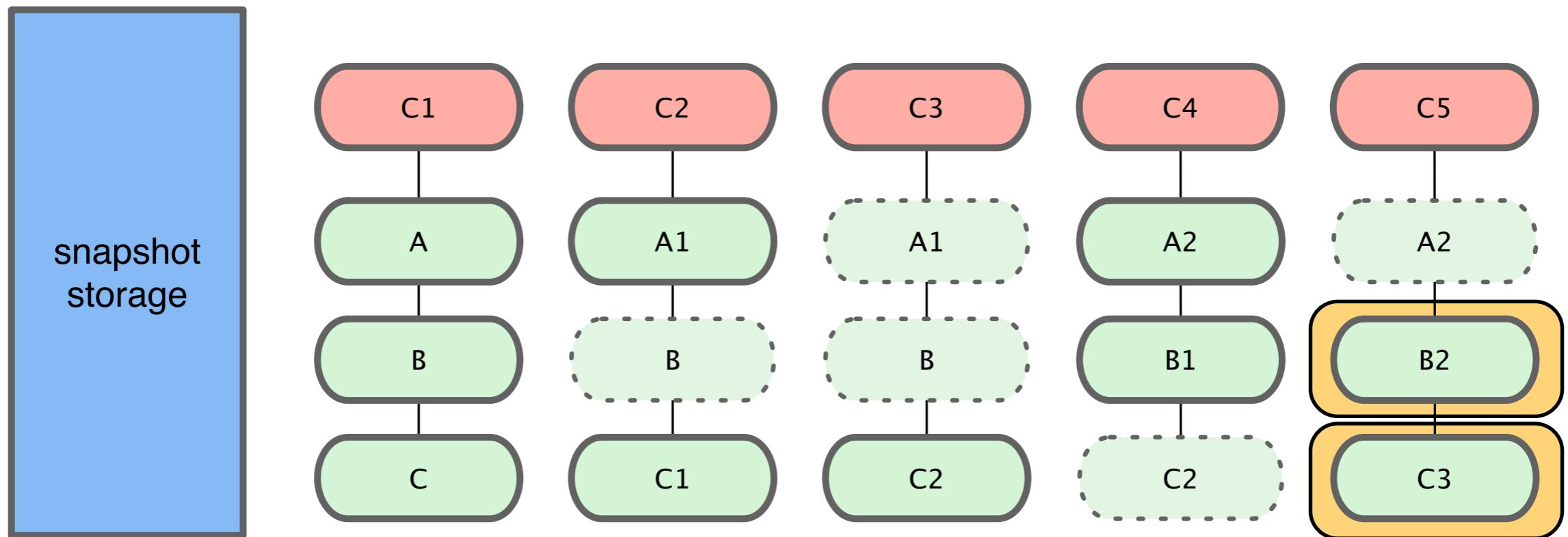
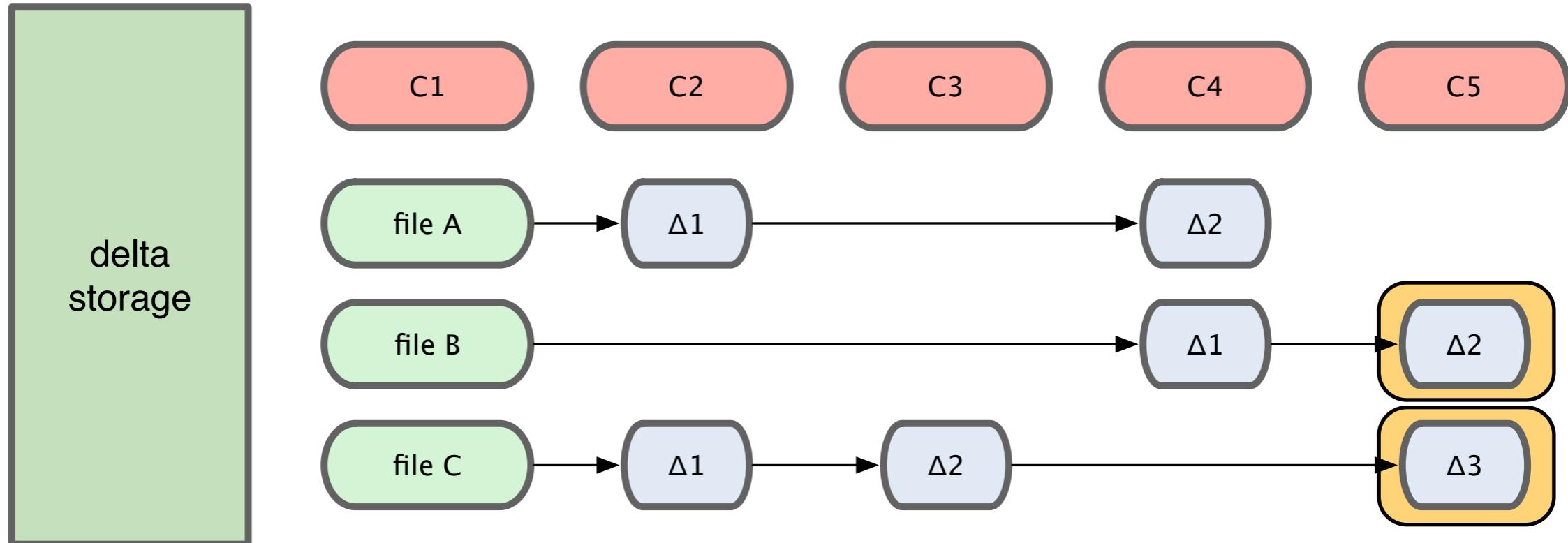


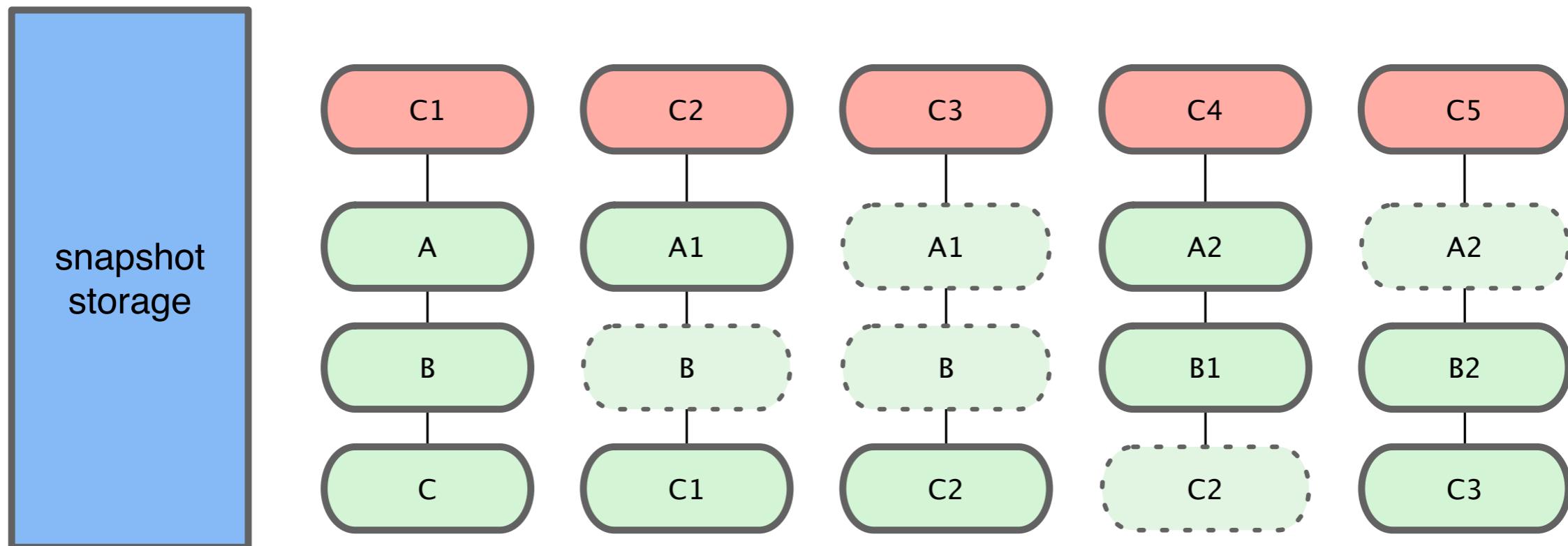
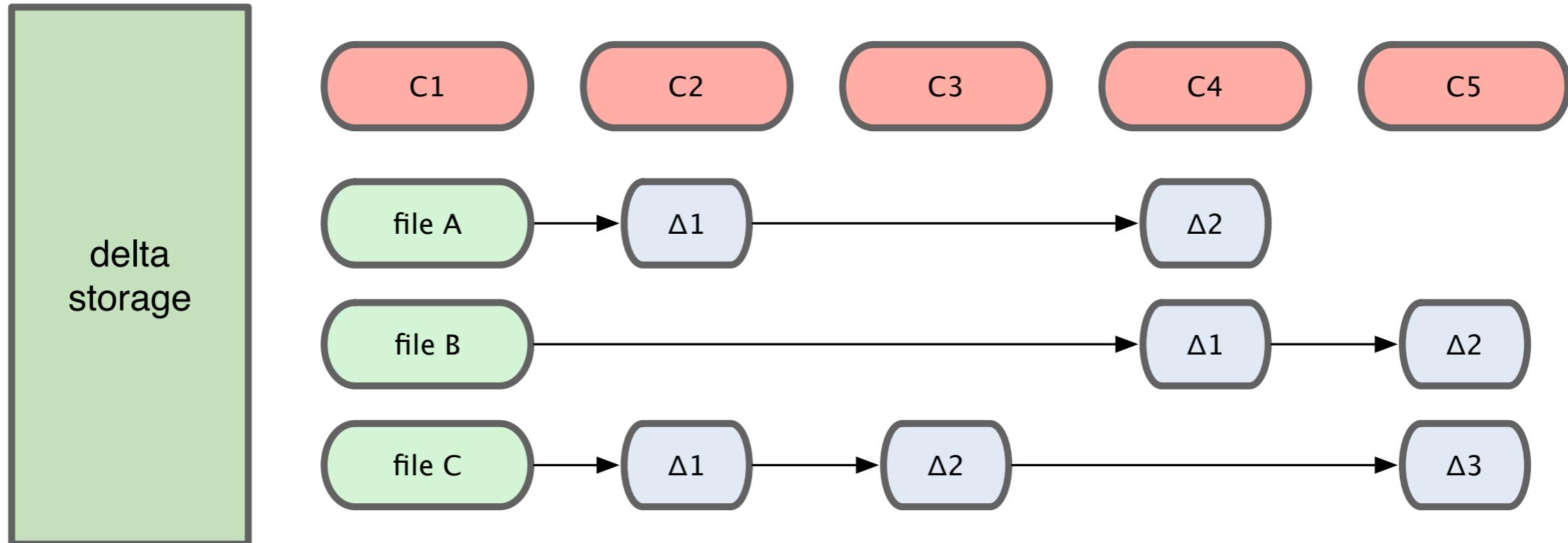


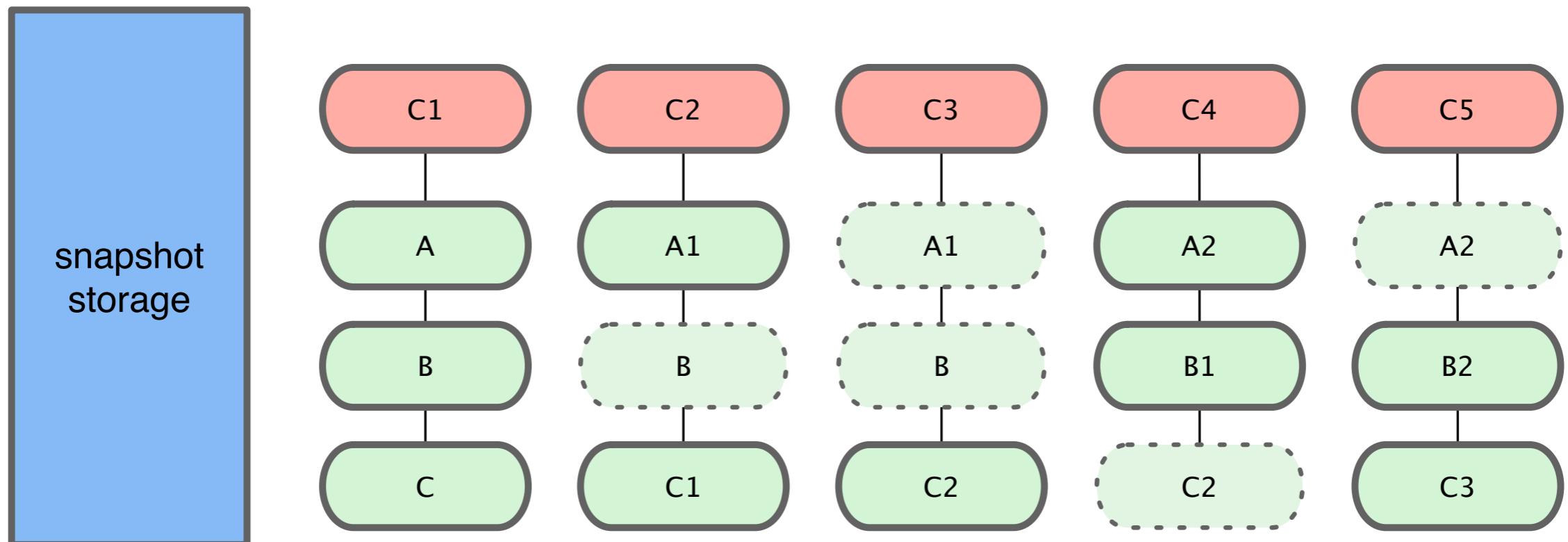
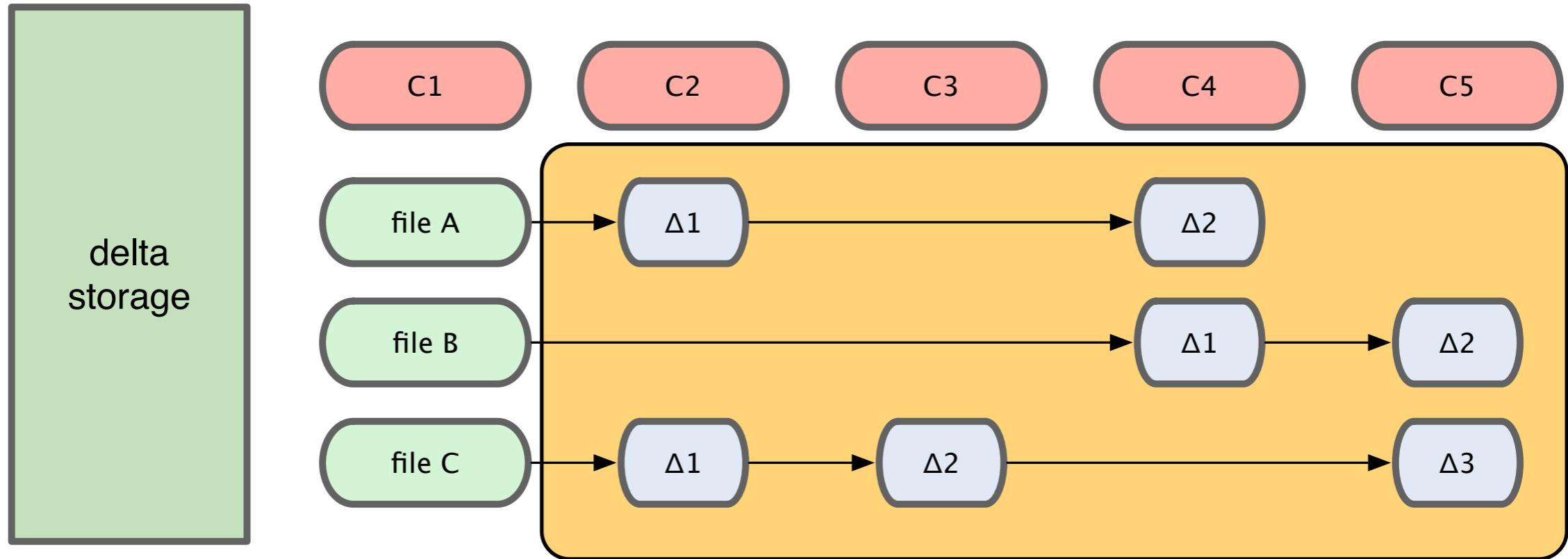


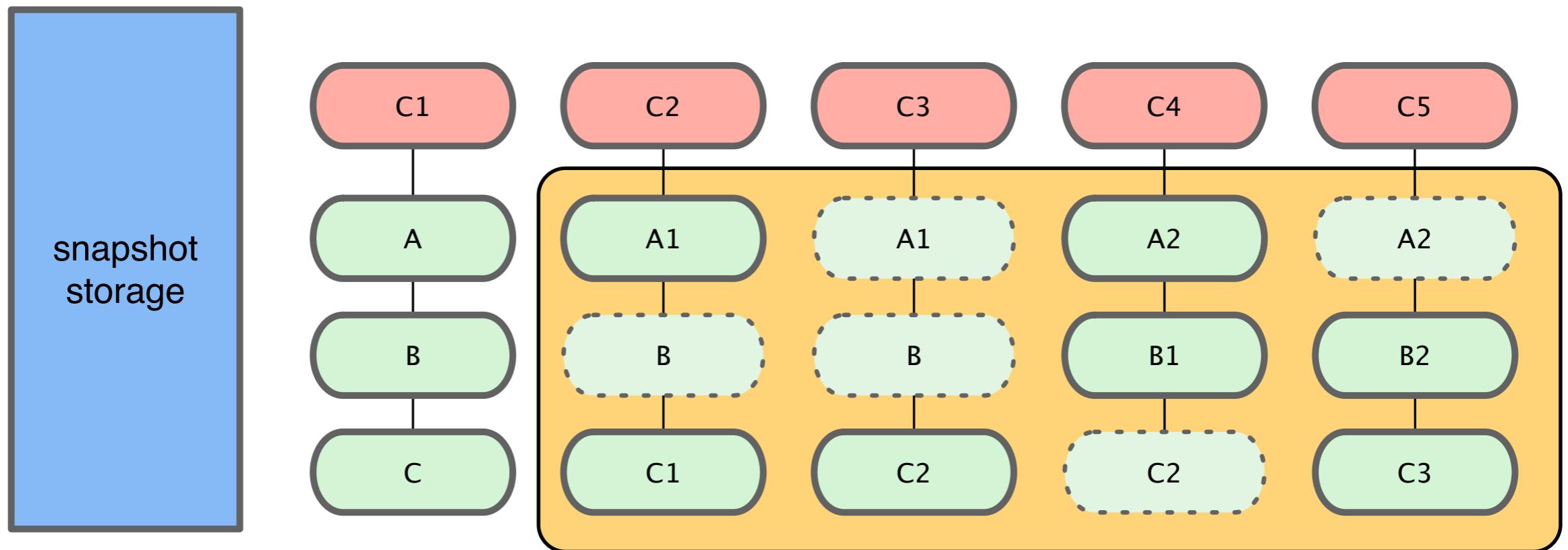
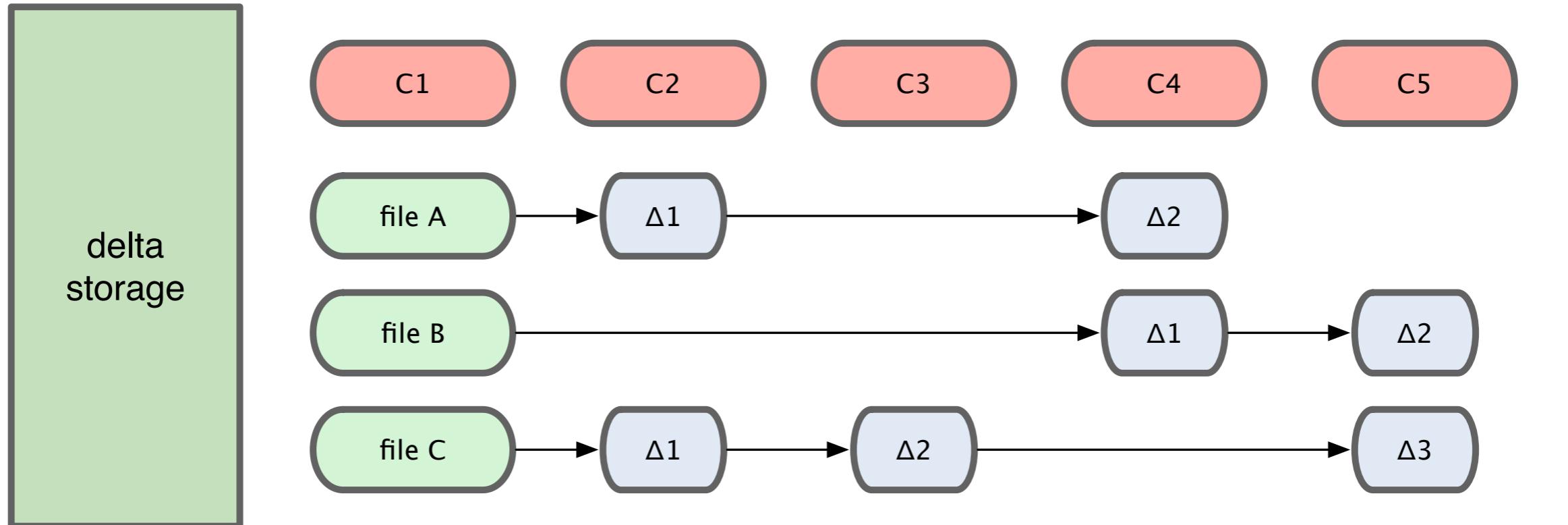














NATIONAL
GEOGRAPHIC

Photograph by Melissa Brandts, Your Shot

© COPYRIGHT NATIONAL GEOGRAPHIC SOCIETY. ALL RIGHTS RESERVED.

Basic Git

First Steps

```
$ git config --global user.name "Scott Chacon"  
$ git config --global user.email "schacon@gmail.com"
```

Getting a Repo

Create One

git init

```
$ touch hello_world.rb
```

```
$ touch hello_world.rb  
$ git init
```

```
$ tree -a  
.  
|-- .git  
|   |-- HEAD  
|   |-- branches  
|   |-- config  
|   |-- description  
|   |-- hooks  
|   |   |-- post-commit.sample  
|   |   |-- post-receive.sample  
|   |   |-- ...  
|   |   |-- pre-rebase.sample  
|   |   `-- update.sample  
|   |-- info  
|   |   `-- exclude  
|   |-- objects  
|   |   |-- info  
|   |   `-- pack  
|   |-- refs  
|   |   |-- heads  
|   |   `-- tags  
|   `-- remotes  
`-- hello_world.rb
```

11 directories, 25 files

```
$ tree -a
```

```
.
|--- .git
|   |--- HEAD
|   |--- branches
|   |--- config
|   |--- description
|   |--- hooks
|   |   |--- post-commit.sample
|   |   |--- post-receive.sample
|   |   |--- ...
|   |   |--- pre-rebase.sample
|   |   `--- update.sample
|   |--- info
|   |   `--- exclude
|   |--- objects
|   |   |--- info
|   |   |--- pack
|   |--- refs
|   |   |--- heads
|   |   |--- tags
|   |--- remotes
`--- hello_world.rb
```

11 directories, 25 files

```
$ tree -a  
.  
|-- .git  
|   |-- HEAD  
|   |-- branches  
|   |-- config  
|   |-- description  
|   |-- hooks  
|   |   |-- post-commit.sample  
|   |   |-- post-receive.sample  
|   |   |-- ...  
|   |   |-- pre-rebase.sample  
|   |   `-- update.sample  
|   |-- info  
|   |   `-- exclude  
|   |-- objects  
|   |   |-- info  
|   |   `-- pack  
|   |-- refs  
|   |   |-- heads  
|   |   `-- tags  
|   `-- remotes  
`-- hello_world.rb
```

11 directories, 25 files

```
$ touch hello_world.rb  
$ git init  
$ git add .
```

```
$ touch hello_world.rb
$ git init
$ git add .
$ git commit -m 'first commit'
```

```
$ tree -a
.
|-- .git
|   |-- COMMIT_EDITMSG
|   |-- HEAD
|   |-- branches
|   |-- config
|   |-- description
|   |-- hooks
|   |   |-- applypatch-msg.sample
|   |   `-- update.sample
|   |-- index
|   |-- info
|   |   '-- exclude
|   |-- logs
|   |   '-- HEAD
|   |   `-- refs
|   |       '-- heads
|   |           '-- master
|   '-- objects
|       '-- 32/09658ac8d80bc9726d3a33d77e3dfc5fe6035e
|       '-- 53/9cd7886a627841d525a78d45cbc6396be20b41
|       '-- e6/9de29bb2d1d6434b8b29ae775ad8c2e48c5391
|       '-- info
|           '-- pack
|   '-- refs
|       '-- heads
|           '-- master
|       '-- tags
|   '-- remotes
`-- hello_world.rb
```

17 directories, 33 files

```
$ tree -a
```

```
.
```

```
|- -- .git
|   |- -- COMMIT_EDITMSG
|   |- -- HEAD
|   |- -- branches
|   |- -- config
|   |- -- description
|   |- -- hooks
|       |- -- applypatch-msg.sample
|       `-- update.sample
|   |- -- index
|   |- -- info
|       `-- exclude
|   |- -- logs
|       |- -- HEAD
|       `-- refs
|           `-- heads
|               `-- master
|   |- -- objects
|       |- -- 32/09658ac8d80bc9726d3a33d77e3dfc5fe6035e
|       |- -- 53/9cd7886a627841d525a78d45cbc6396be20b41
|       |- -- e6/9de29bb2d1d6434b8b29ae775ad8c2e48c5391
|       |- -- info
|       `-- pack
|   |- -- refs
|       |- -- heads
|           `-- master
|       `-- tags
|   `-- remotes
`-- hello_world.rb
```

17 directories, 33 files

```
$ tree -a
.
|-- .git
|   |-- COMMIT_EDITMSG
|   |-- HEAD
|   |-- branches
|   |-- config
|   |-- description
|   |-- hooks
|       |-- applypatch-msg.sample
|       `-- update.sample
|   |-- index
|   |-- info
|       `-- exclude
|   |-- logs
|       |-- HEAD
|       `-- refs
|           `-- heads
|               `-- master
|-- objects
|   |-- 32/09658ac8d80bc9726d3a33d77e3dfc5fe6035e
|   |-- 53/9cd7886a627841d525a78d45cbc6396be20b41
|   |-- e6/9de29bb2d1d6434b8b29ae775ad8c2e48c5391
|   |-- info
|       `-- pack
|-- refs
|   |-- heads
|       `-- master
|   `-- tags
`-- remotes
`-- hello_world.rb
```

17 directories, 33 files

Clone One

git clone

```
$ git clone git://github.com/schacon/ticgit.git
```

```
$ git clone git://github.com/schacon/ticgit.git
Initialized empty Git repository in /private/tmp/ticgit/.git/
remote: Counting objects: 591, done.
remote: Compressing objects: 100% (267/267), done.
remote: Total 591 (delta 253), reused 587 (delta 252)
Receiving objects: 100% (591/591), 73.05 KiB, done.
Resolving deltas: 100% (253/253), done.
$
```

```
$ git clone git://github.com/schacon/ticgit.git
Initialized empty Git repository in /private/tmp/ticgit/.git/
remote: Counting objects: 591, done.
remote: Compressing objects: 100% (267/267), done.
remote: Total 591 (delta 253), reused 587 (delta 252)
Receiving objects: 100% (591/591), 73.05 KiB, done.
Resolving deltas: 100% (253/253), done.
$ cd ticgit/
$
```

```
$ git clone git://github.com/schacon/ticgit.git
Initialized empty Git repository in /private/tmp/ticgit/.git/
remote: Counting objects: 591, done.
remote: Compressing objects: 100% (267/267), done.
remote: Total 591 (delta 253), reused 587 (delta 252)
Receiving objects: 100% (591/591), 73.05 KiB, done.
Resolving deltas: 100% (253/253), done.
$ cd ticgit/
$ ls
.git  LICENSE      Rakefile  examples  note
ticgit.gemspec    README     bin       lib      spec
$
```

A Basic Workflow

A Basic Workflow

Edit files

Stage the changes

Review your changes

Commit the changes

working directory

index

repository

working directory

a working copy
of your project

index

repository

working directory

index

object database

repository

working directory

index

“staging”

repository

A Basic Workflow

Edit files

Stage the changes

Review your changes

Commit the changes

\$

\$

```
$ vim main.py
```

```
#!/usr/bin/env python
import wsgiref.handlers
from google.appengine.ext import webapp

# this program prints out 'hello world'

class MainHandler(webapp.RequestHandler):

    def get(self):
        self.response.out.write('Hello world!')

    def main():
        application = webapp.WSGIApplication([(('/', MainHandler)],
                                              debug=True)
        wsgiref.handlers.CGIHandler().run(application)

    if __name__ == '__main__':
        main()
~  
~  
"main.py" 16L, 402C
```

```
$ git status
```

```
$ git status
```

```
# On branch master
```

```
# Changed but not updated:
```

```
#   (use "git add <file>..." to update what
```

```
#
```

```
# modified:    main.py
```

```
#
```

```
no changes added to commit (use "git add" and
```

```
$ git status
```

```
# On branch master
```

```
# Changed but not updated:
```

```
#   (use "git add <file>..." to update what
```

```
#
```

```
# modified:    main.py
```

```
#
```

```
no changes added to commit (use "git add" and
```

```
$ git status
```

```
# On branch master
```

```
# Changed but not updated:
```

```
#   (use "git add <file>..." to update what
```

```
#
```

```
# modified:    main.py
```

```
#
```

```
no changes added to commit (use "git add" and
```

```
$ git status
```

```
# On branch master
```

```
# Changed but not STAGED:
```

```
#       (use "git add <file>..." to update what
```

```
#
```

```
# modified:    main.py
```

```
#
```

```
no changes added to commit (use "git add" and
```

A Basic Workflow

Edit files

Stage the changes

Review your changes

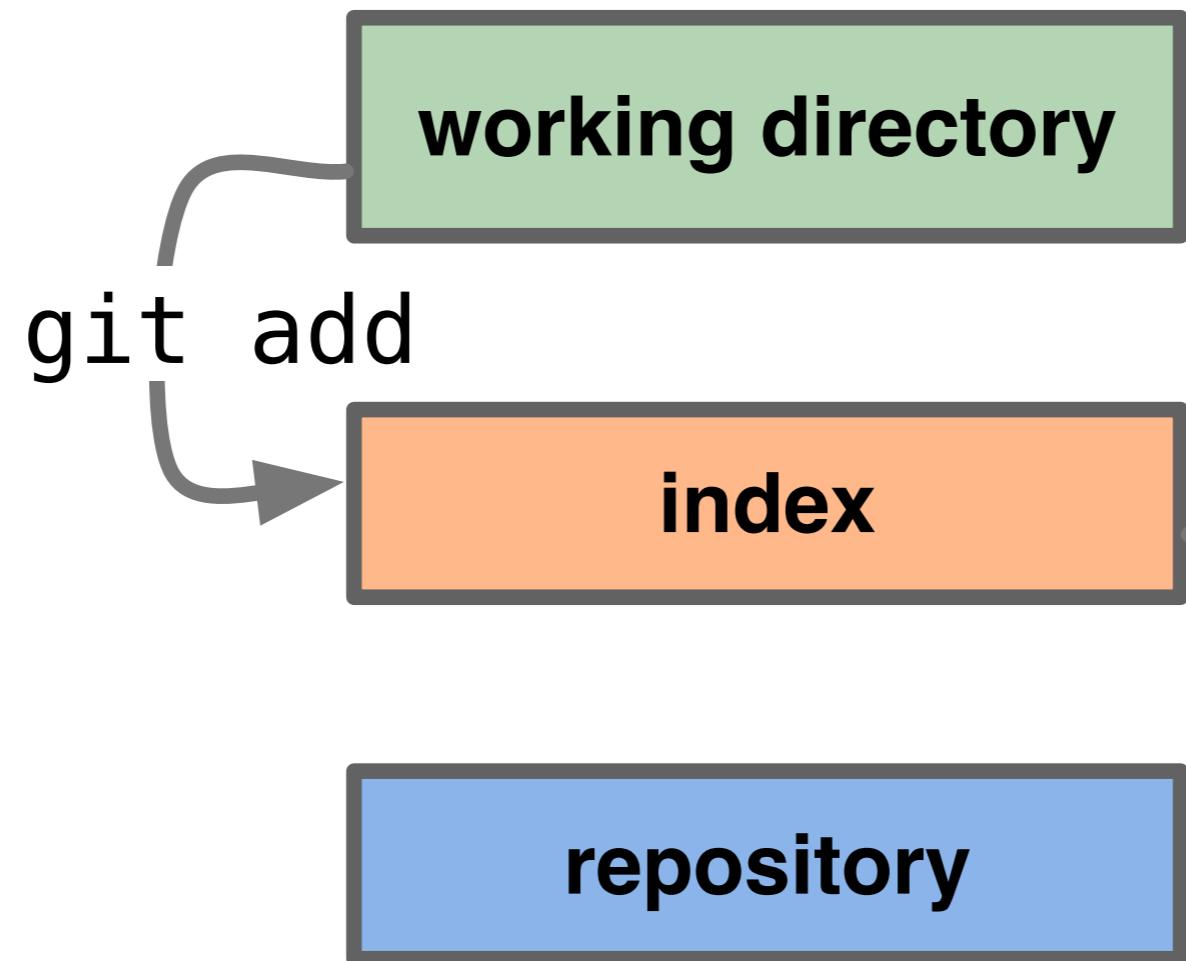
Commit the changes

git add

working directory

index

repository



\$

```
$ git add main.py
$ git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>
#
# modified:    main.py
#
```

```
$ git add main.py
$ git status
# On branch master
# Changes THAT ARE STAGED :
#   (use "git reset HEAD <file>
#
# modified:    main.py
#
```

\$

```
$ vim app.yaml
```

```
application: chacon
version: █
runtime: python
api_version: 1
```

```
handlers:
- url: .*
  script: main.py
```

```
~
```

```
~
```

```
~
```

```
"app.yaml" 8L, 101C
```

```
$ vim app.yaml
```

```
application: chacon
version: 2
runtime: python
api_version: 1
```

```
handlers:
- url: .*
  script: main.py
```

```
~
```

```
~
```

```
~
```

```
"app.yaml" 8L, 101C
```

\$

\$

```
$ vim main.py
```

```
#!/usr/bin/env python
import wsgiref.handlers
from google.appengine.ext import webapp

# this program prints out 'hello world'

class MainHandler(webapp.RequestHandler):

    def get(self):
        self.response.out.write('Hola █ orld!')

def main():
    application = webapp.WSGIApplication([('/', MainHandler)],
                                          debug=True)
    wsgiref.handlers.CGIHandler().run(application)

if __name__ == '__main__':
    main()
~
~
"main.py" 16L, 402C
```

```
$ vim main.py
```

```
#!/usr/bin/env python
import wsgiref.handlers
from google.appengine.ext import webapp

# this program prints out 'hello world'

class MainHandler(webapp.RequestHandler):

    def get(self):
        self.response.out.write('Hola Mundo')

def main():
    application = webapp.WSGIApplication(['/'], MainHandler),
                                         debug=True)
    wsgiref.handlers.CGIHandler().run(application)

if __name__ == '__main__':
    main()
~  
~  
"main.py" 16L, 402C
```

\$

\$ git status

```
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:   main.py
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be com
#
# modified:   app.yaml
# modified:   main.py
#
```

Staged



\$ git status

```
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:   main.py
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be com
#
# modified:   app.yaml
# modified:   main.py
#
```

Unstaged



\$ git status

```
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   main.py
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be com
#
#       modified:   app.yaml
#       modified:   main.py
#
```



Staged

```
#!/usr/bin/env python
import wsgiref.handlers
from google.appengine.ext import webapp

# this program prints out 'hello world'

class MainHandler(webapp.RequestHandler):

    def get(self):
        self.response.out.write('Hello World!')

def main():
    application = webapp.WSGIApplication
    ([('/' +
        debug=True)
    wsgiref.handlers.CGIHandler().run
    applicat

if __name__ == '__main__':
    main()
```

In Working Directory

```
#!/usr/bin/env python
import wsgiref.handlers
from google.appengine.ext import webapp

# this program prints out 'hello world'

class MainHandler(webapp.RequestHandler):

    def get(self):
        self.response.out.write('Hola Mundo!')

def main():
    application = webapp.WSGIApplication
    ([('/' +
        debug=True)
    wsgiref.handlers.CGIHandler().run
    applicat

if __name__ == '__main__':
    main()
```

Staged

```
#!/usr/bin/env python
import wsgiref.handlers
from google.appengine.ext import webapp

# this program prints out 'hello world'

class MainHandler(webapp.RequestHandler):

    def get(self):
        self.response.out.write('Hello World!')

def main():
    application = webapp.WSGIApplication
    ([('/' +
        debug=True)
    wsgiref.handlers.CGIHandler().run
    applicat

if __name__ == '__main__':
    main()
```

In Working Directory

```
#!/usr/bin/env python
import wsgiref.handlers
from google.appengine.ext import webapp

# this program prints out 'hello world'

class MainHandler(webapp.RequestHandler):

    def get(self):
        self.response.out.write('Hola Mundo!')

def main():
    application = webapp.WSGIApplication
    ([('/' +
        debug=True)
    wsgiref.handlers.CGIHandler().run
    applicat

if __name__ == '__main__':
    main()
```

You have to stage a file
after you edit it

You have to stage a file
after you edit it

You have to stage a file
after you edit it

\$

A Basic Workflow

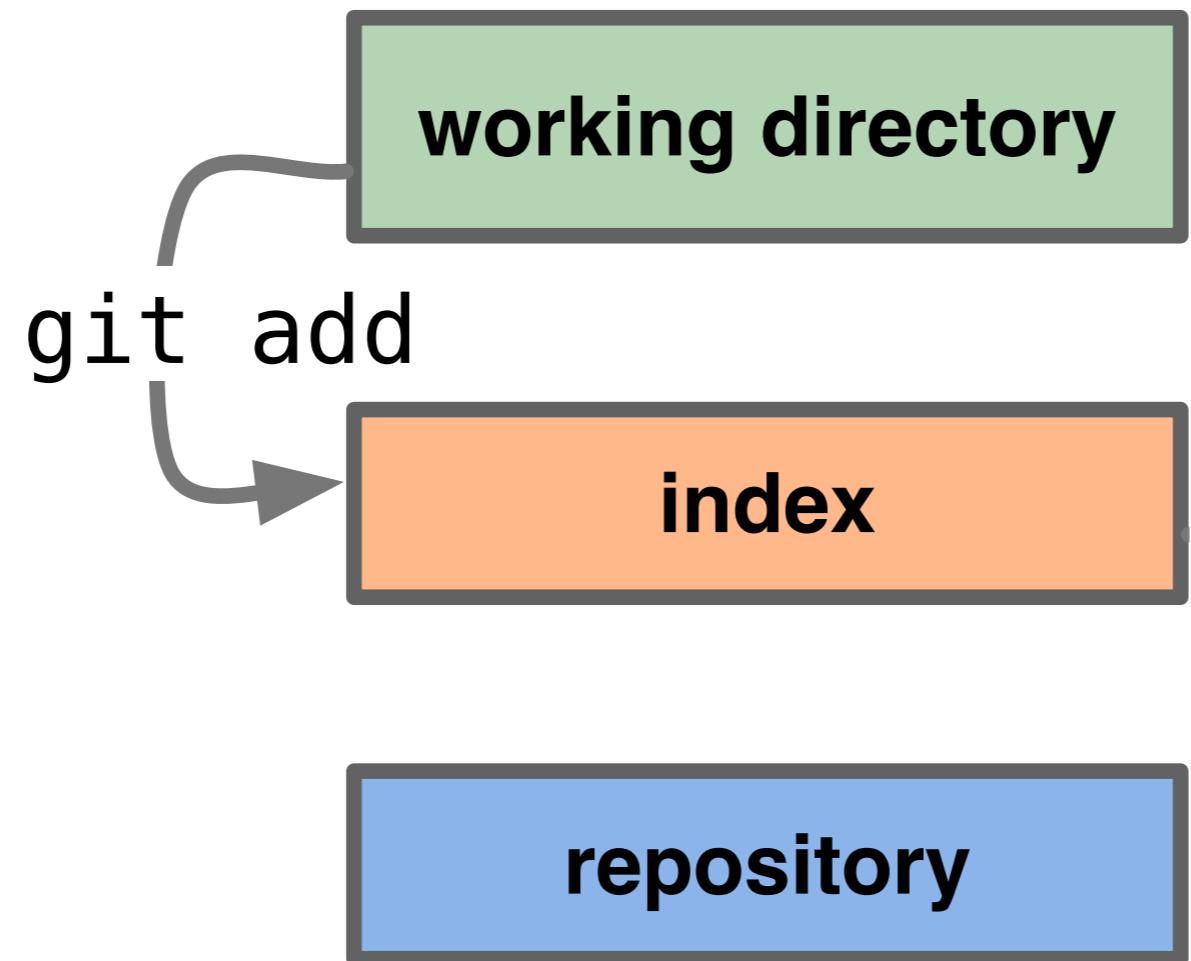
Edit files

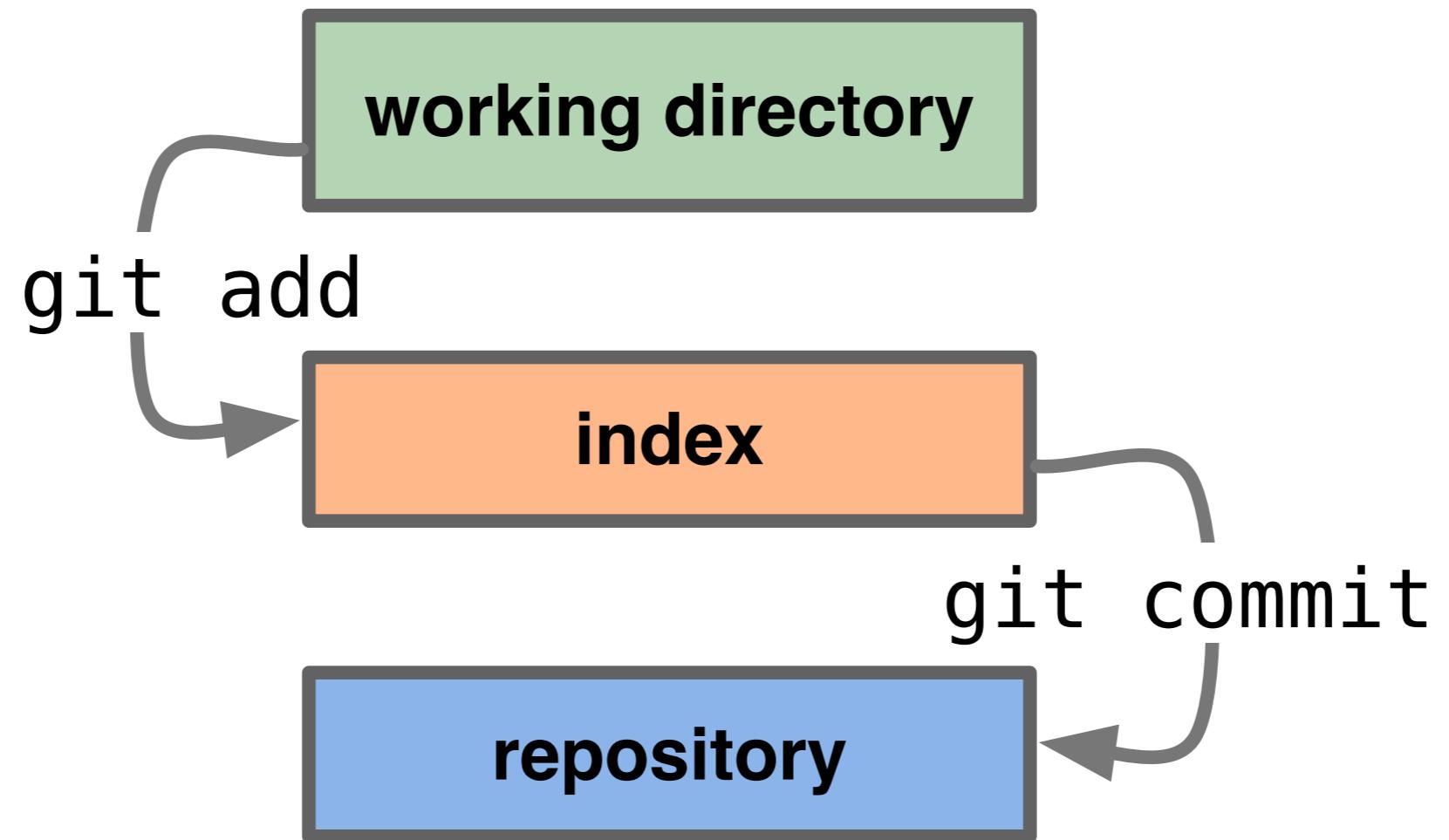
Stage the changes

Review your changes

Commit the changes

git commit





\$

\$ git commit

```
descriptive commit message █  
# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the commit.  
# On branch master  
# Changes to be committed:  
#   (use "git reset HEAD <file>..." to unstage)  
#  
#       modified:   app.yaml  
#       modified:   main.py  
#  
~  
~  
~  
~  
~  
".git/COMMIT_EDITMSG" 10L, 279C
```

\$ git commit

```
descriptive commit message █  
# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the commit.  
# On branch master  
# Changes to be committed:  
#   (use "git reset HEAD <file>..." to unstage)  
#  
#       modified:   app.yaml  
#       modified:   main.py  
#  
~  
~  
~  
~  
~  
".git/COMMIT_EDITMSG" 10L, 279C
```

```
$ git commit
```

```
Created commit 77d3001: descriptive commit message  
2 files changed, 4 insertions(+), 2 deletions(-)
```

A Basic Workflow

A Basic Workflow

Edit files

vim / emacs / etc

A Basic Workflow

Edit files

`vim / emacs / etc`

Stage the changes

`git add (file)`

A Basic Workflow

Edit files

`vim / emacs / etc`

Stage the changes

`git add (file)`

Review your changes

`git status`

A Basic Workflow

Edit files

`vim / emacs / etc`

Stage the changes

`git add (file)`

Review your changes

`git status`

Commit the changes

`git commit`

A Basicer Workflow

Edit files

`vim / emacs / etc`

Stage and Commit

`git commit -a`

What's going on here?

```
$ git commit
```

```
Created commit 77d3001: descriptive commit message  
2 files changed, 4 insertions(+), 2 deletions(-)
```

```
$ git commit
```

```
Created commit 77d3001: descriptive commit message  
2 files changed, 4 insertions(+), 2 deletions(-)
```

77d3001

77d3001a1de6bf8f5e431972fe4d25b01e595c0b

77d3001a1de6bf8f5e431972fe4d25b01e595c0b

77d3001a1de6bf8f5e431972fe4d25b01e595c0b

commit	size
tree	c4ec5
parent	a149e
author	Scott
committer	Scott
my commit message goes here and it is really, really cool	

77d3001a1de6bf8f5e431972fe4d25b01e595c0b

```
tree c4ec543b0322744e55c5efc9b6c4e449d398dbff
parent a149e2160b3f7573768cdc2fce24d0881f3577e1
author Scott Chacon <schacon@gmail.com> 1223402504 -0700
committer Scott Chacon <schacon@gmail.com> 1223402504 -0700
```

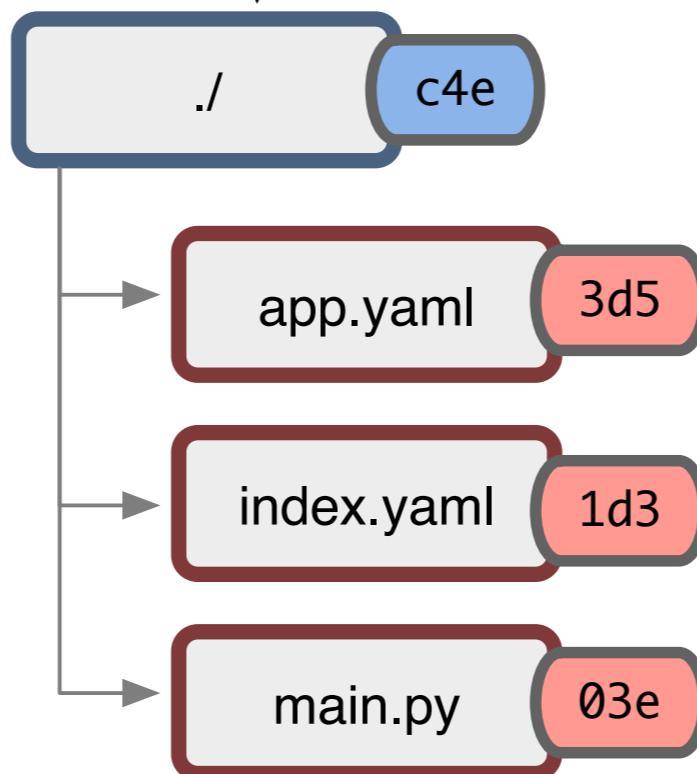
descriptive commit message

77d3001a1de6bf8f5e431972fe4d25b01e595c0b

commit	size
tree	c4ec5
parent	a149e
author	Scott
committer	Scott
my commit message goes here and it is really, really cool	

77d3001a1de6bf8f5e431972fe4d25b01e595c0b

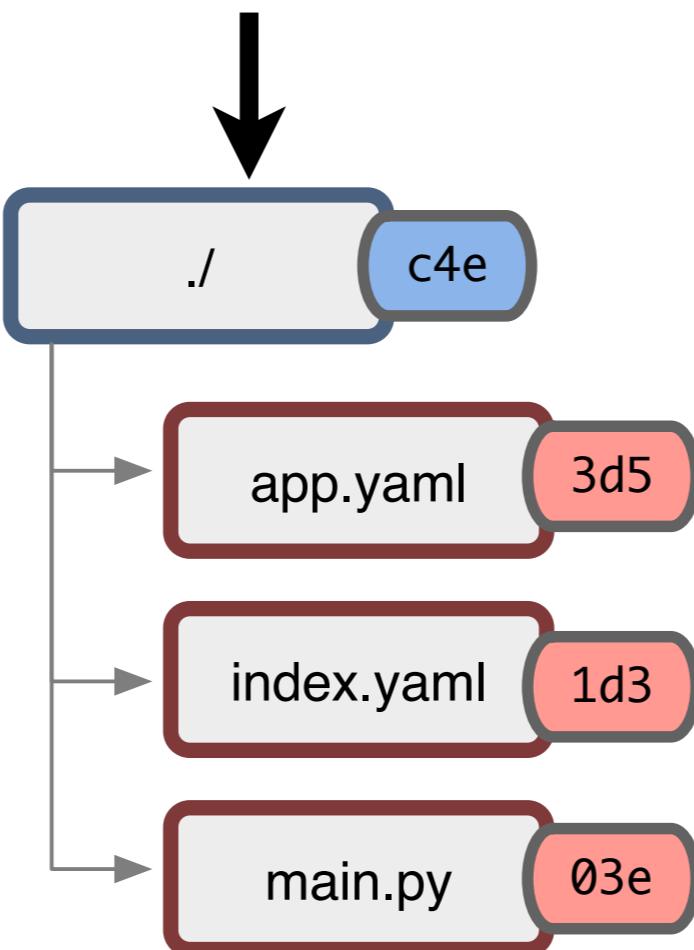
commit	size
tree	c4ec5
parent	a149e
author	Scott
committer	Scott
my commit message goes here and it is really, really cool	



commit	size
tree	2de54
parent	38def
author	Scott
committer	Scott
this is the previous commit and I am very proud of it	

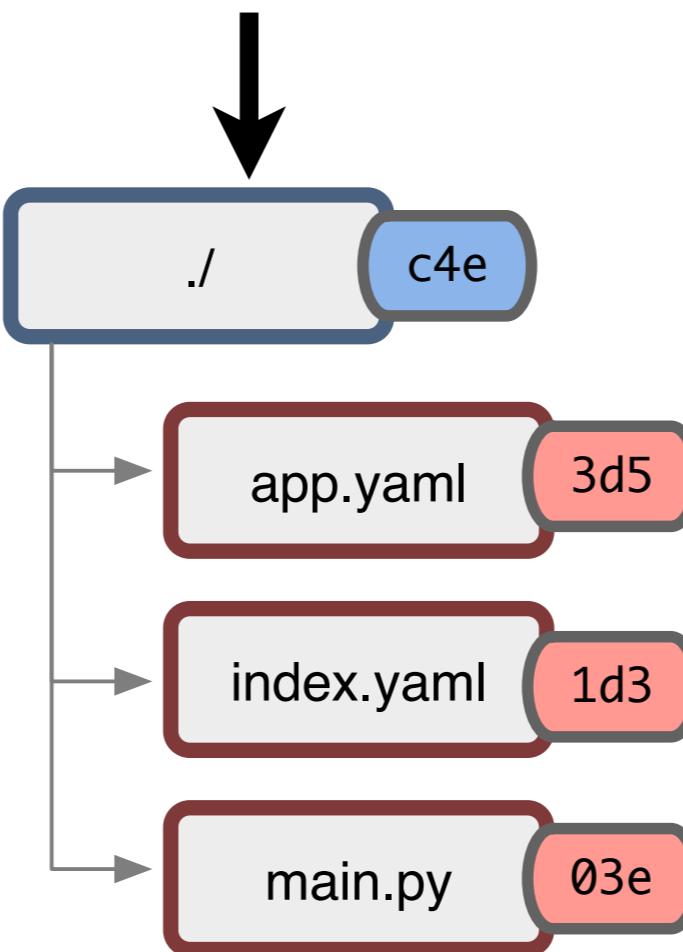
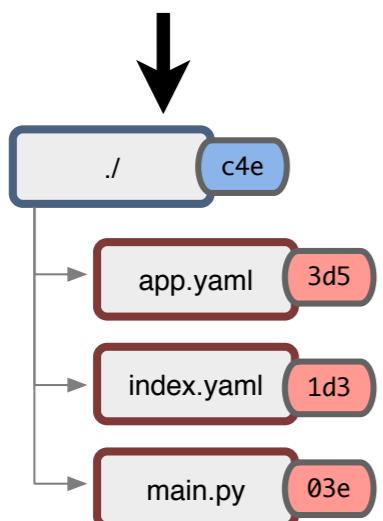
←

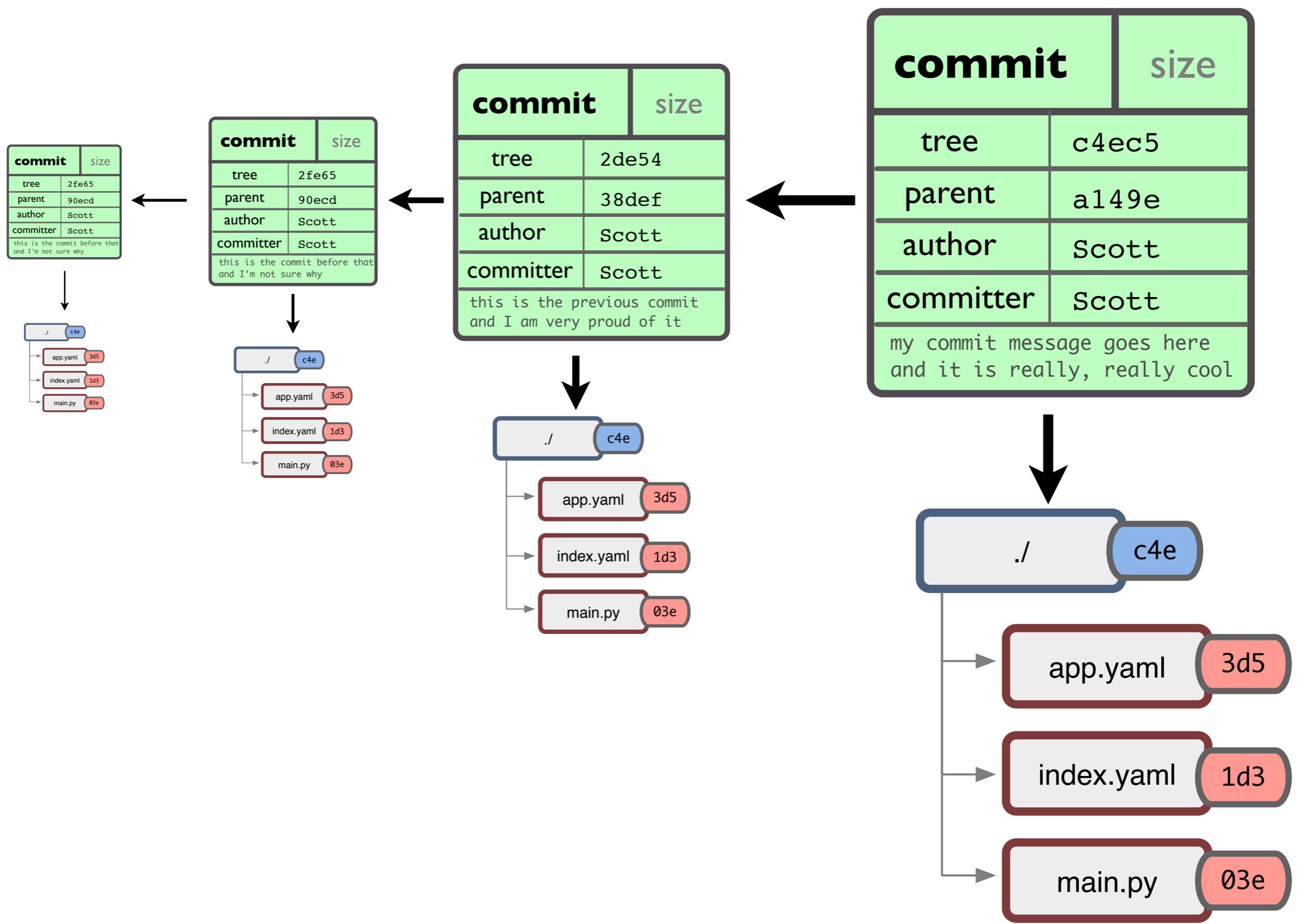
commit	size
tree	c4ec5
parent	a149e
author	Scott
committer	Scott
my commit message goes here and it is really, really cool	

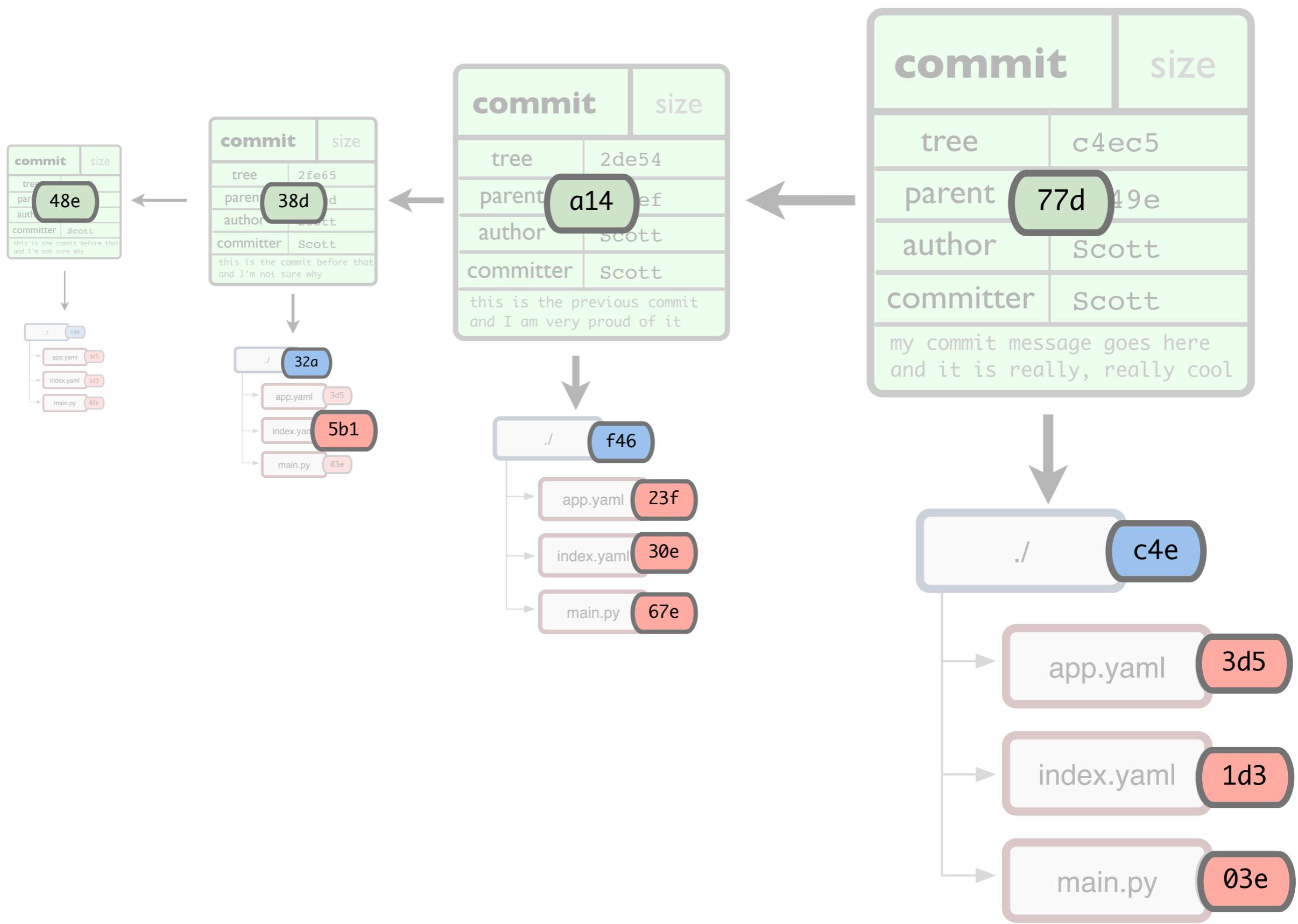


commit	size
tree	2de54
parent	38def
author	Scott
committer	Scott
this is the previous commit and I am very proud of it	

commit	size
tree	c4ec5
parent	a149e
author	Scott
committer	Scott
my commit message goes here and it is really, really cool	







Repository

5b1	32a	c36
1d3	f46	3d4
ffe	6fe	ae6
38d	23f	03e
254	30e	5b1
a14	67e	1d3
3d5	735	d23
c4e	c4e	48e
77d	de3	2d3

Repository

5b1	32a	c36
1d3	f46	3d4
ffe	6fe	ae6
38d	23f	03e
254	30e	5b1
a14	67e	1d3
3d5	735	d23
c4e	c4e	48e
77d	de3	2d3

git checkout ae635

Repository

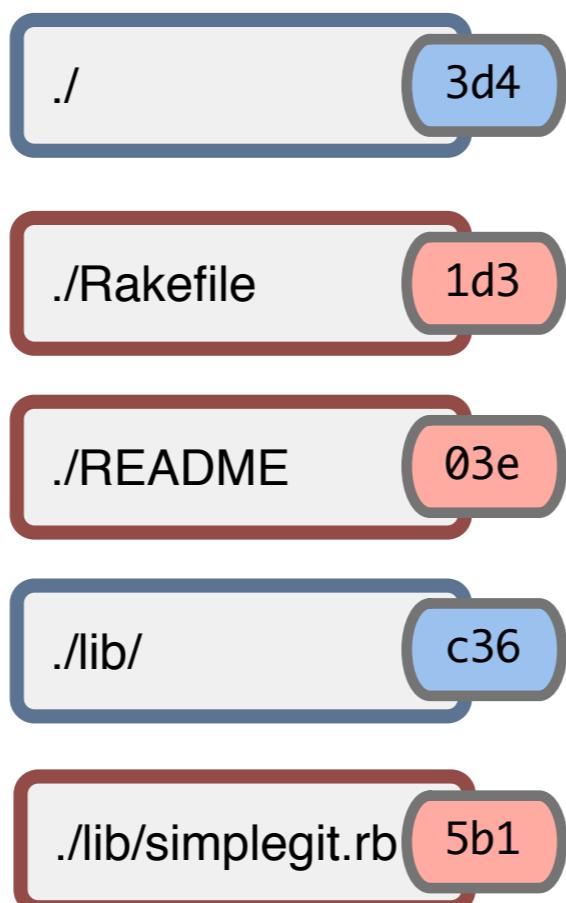
5b1	32a	c36
1d3	f46	3d4
ffe	6fe	ae6
38d	23f	03e
254	30e	5b1
a14	67e	1d3
3d5	735	d23
c4e	c4e	48e
77d	de3	2d3

git checkout ae635

Repository



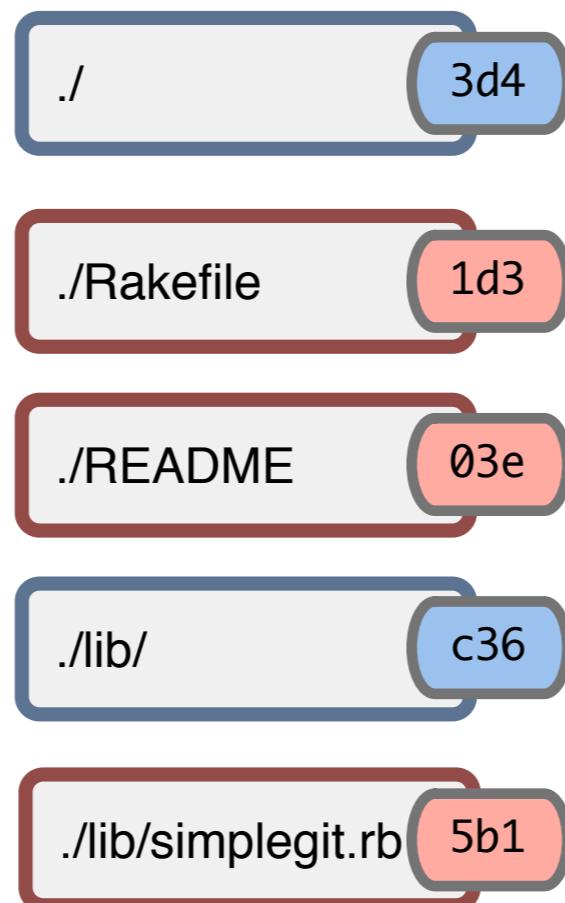
Index



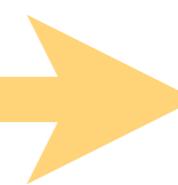
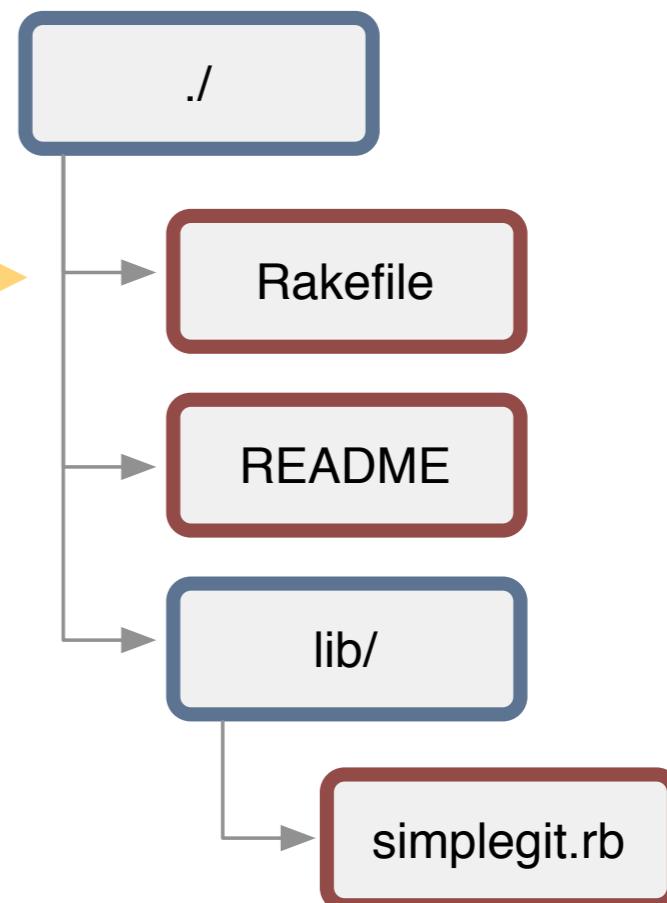
Repository



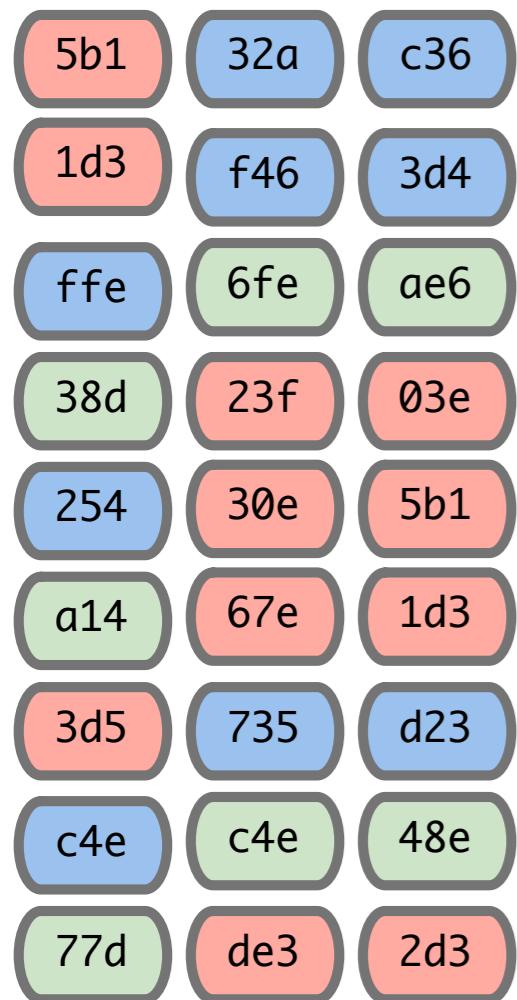
Index



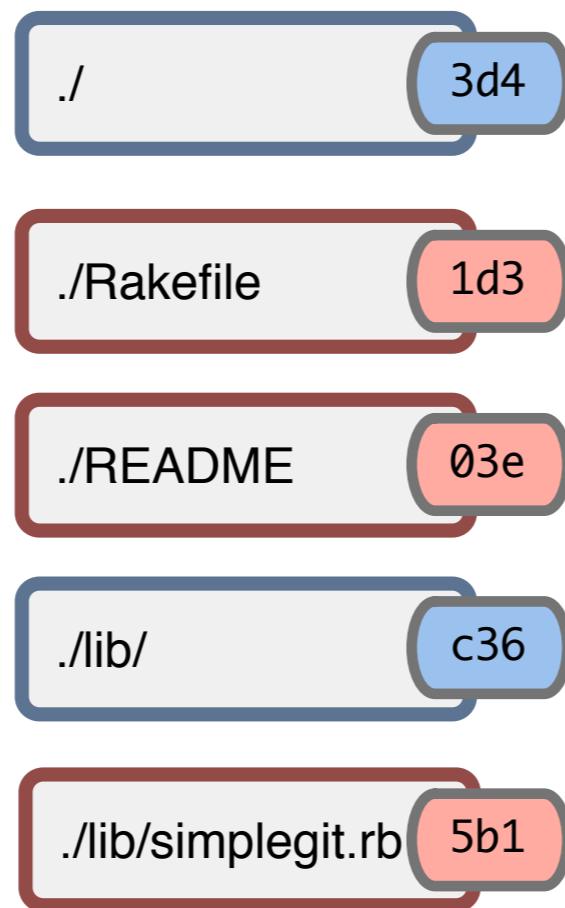
Working Directory



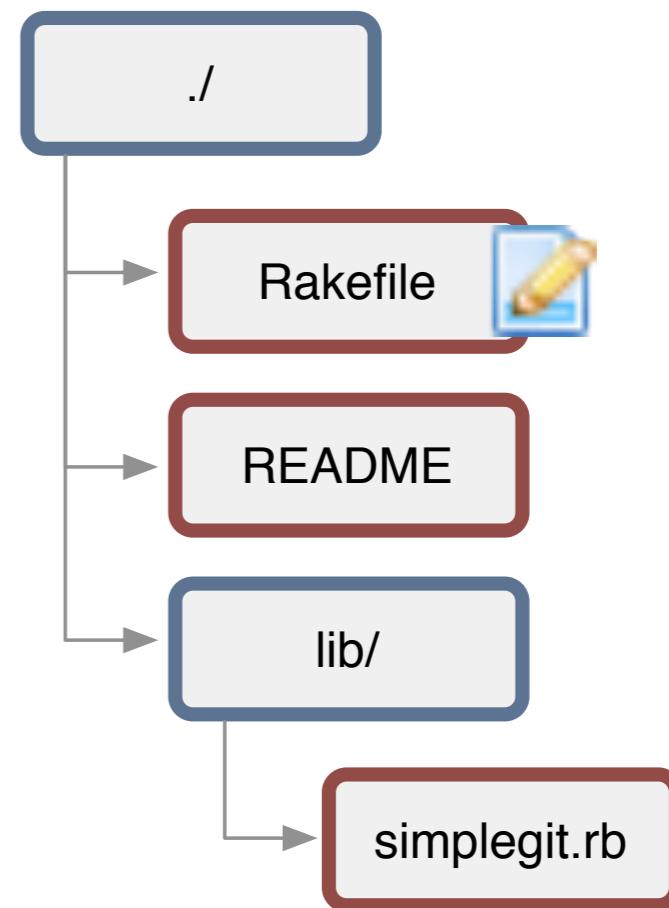
Repository



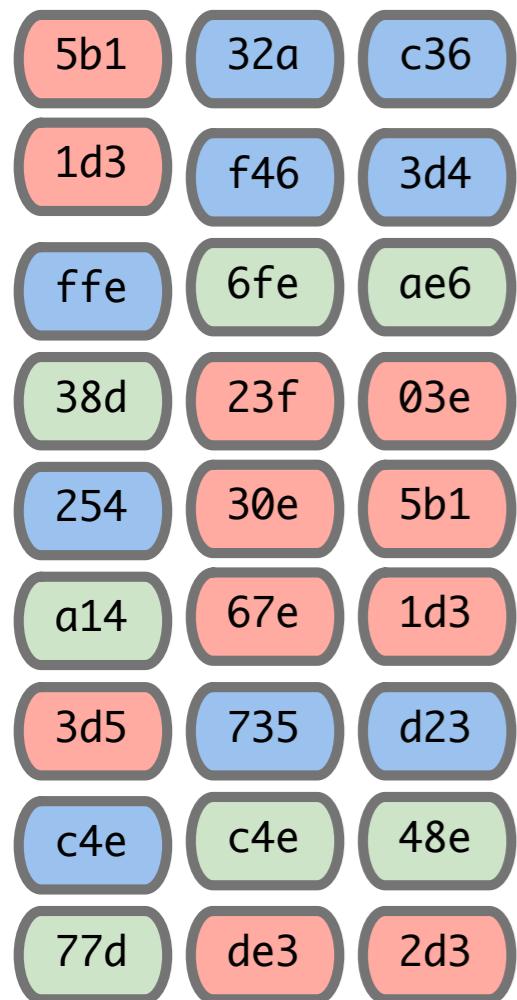
Index



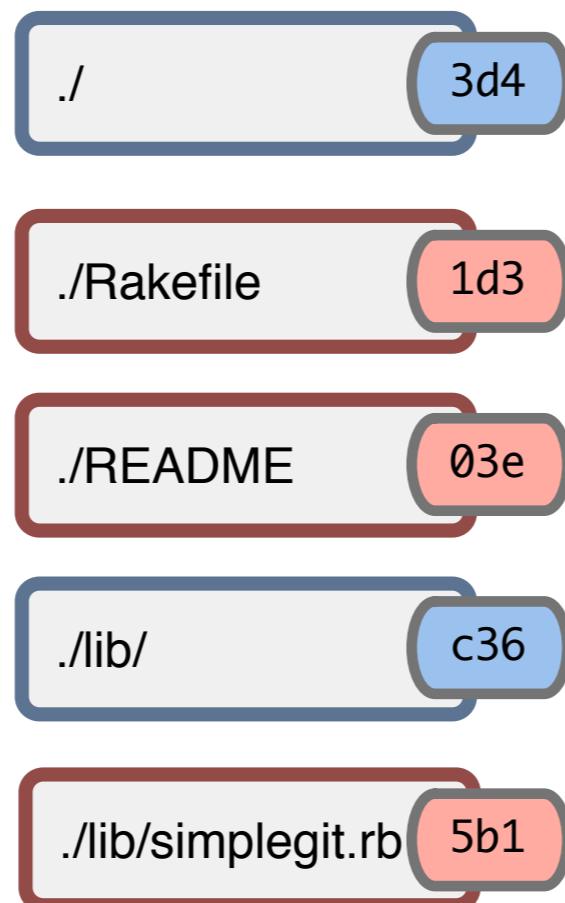
Working Directory



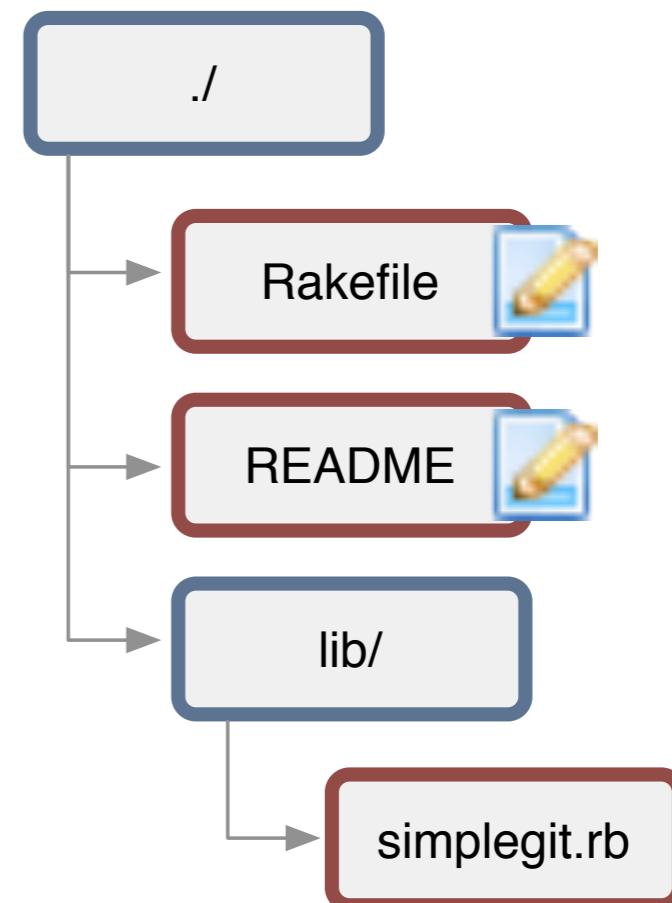
Repository



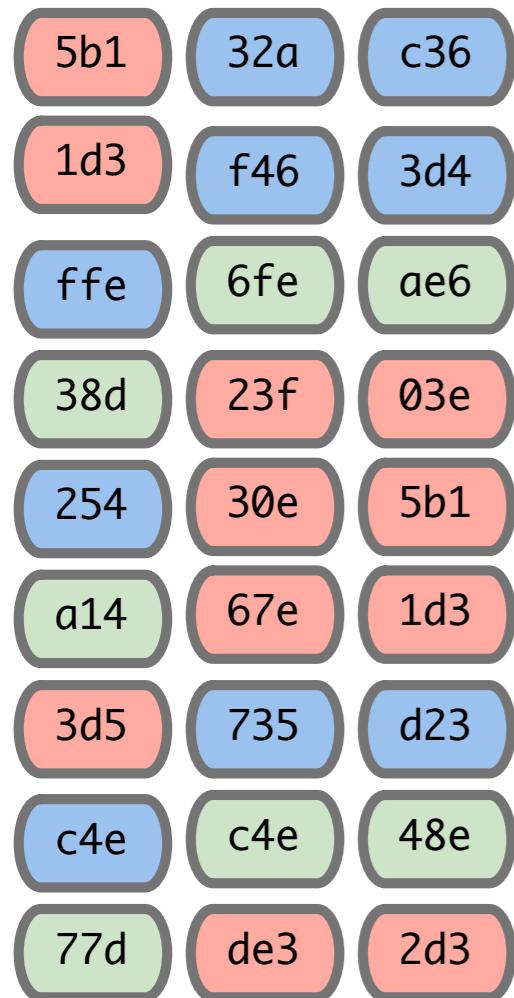
Index



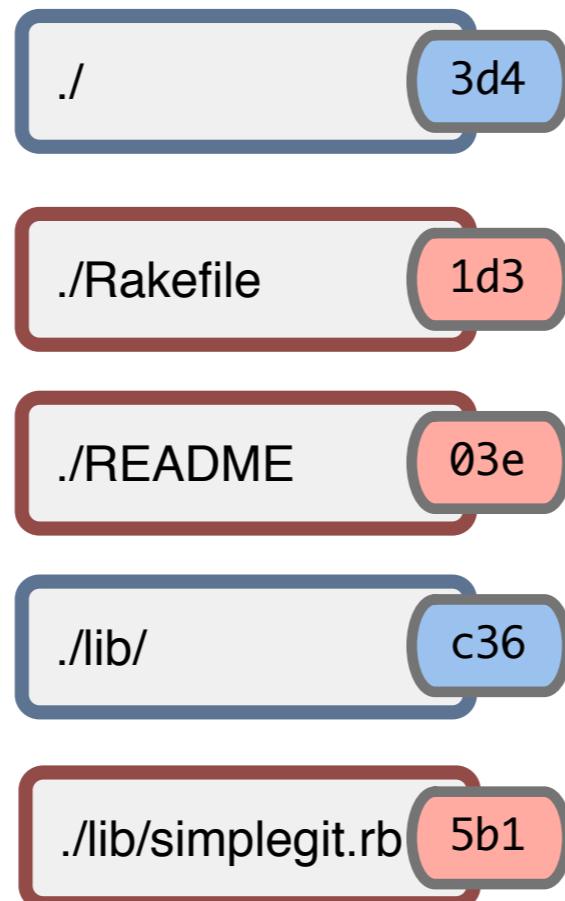
Working Directory



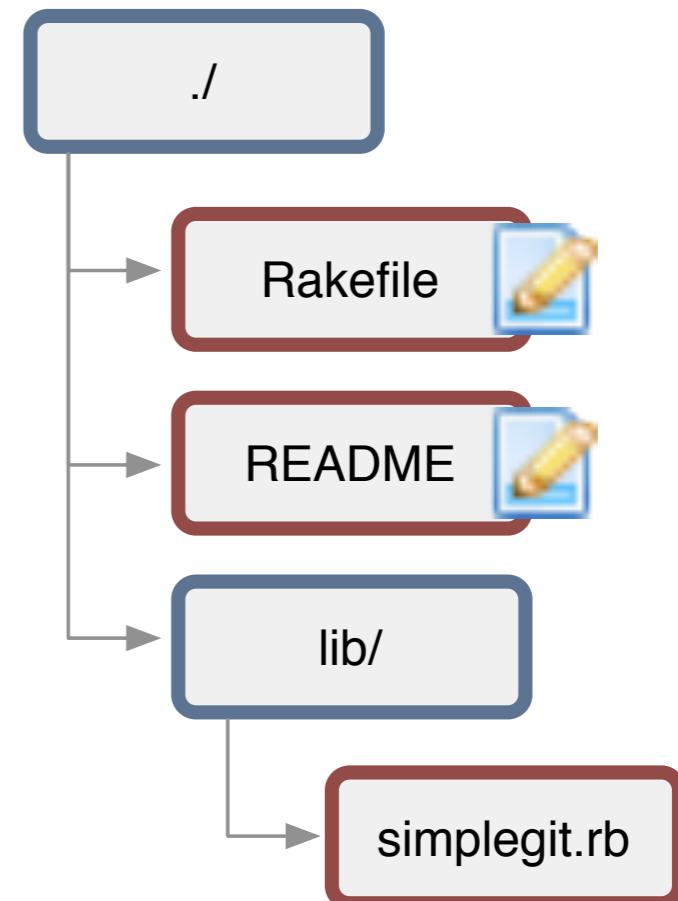
Repository



Index

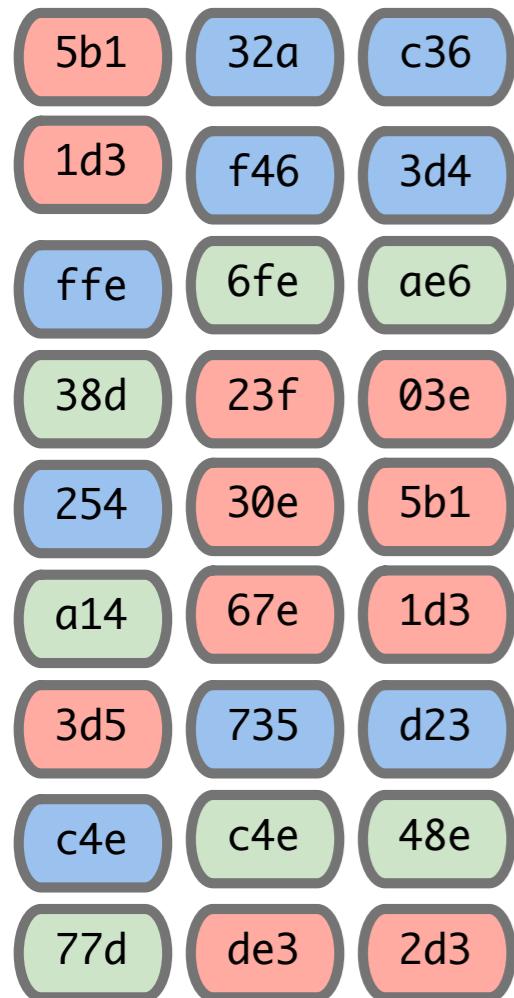


Working Directory

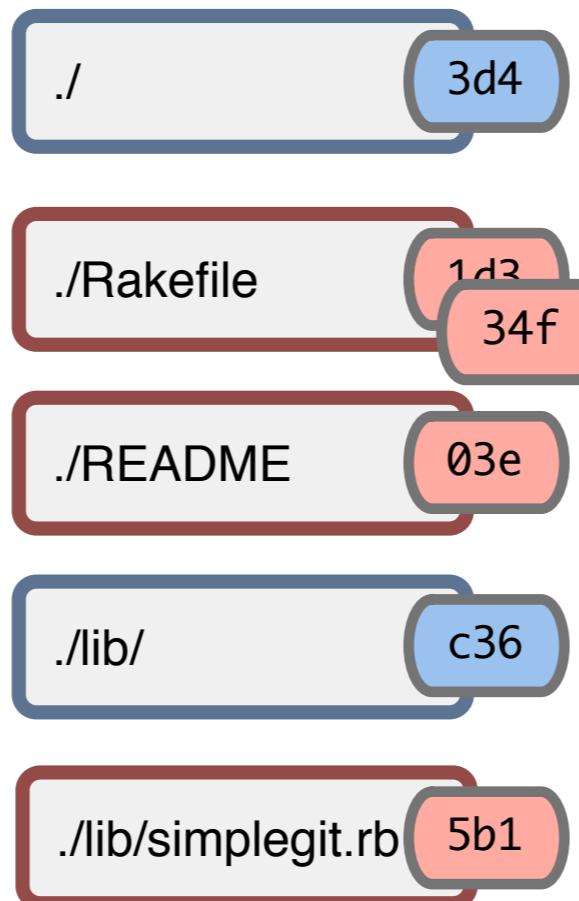


git add

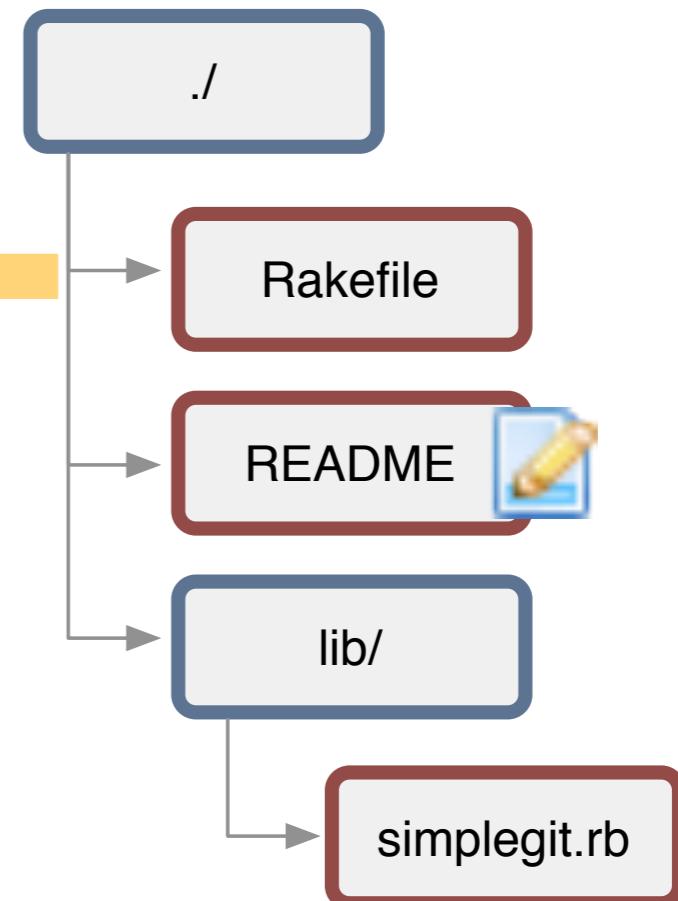
Repository



Index

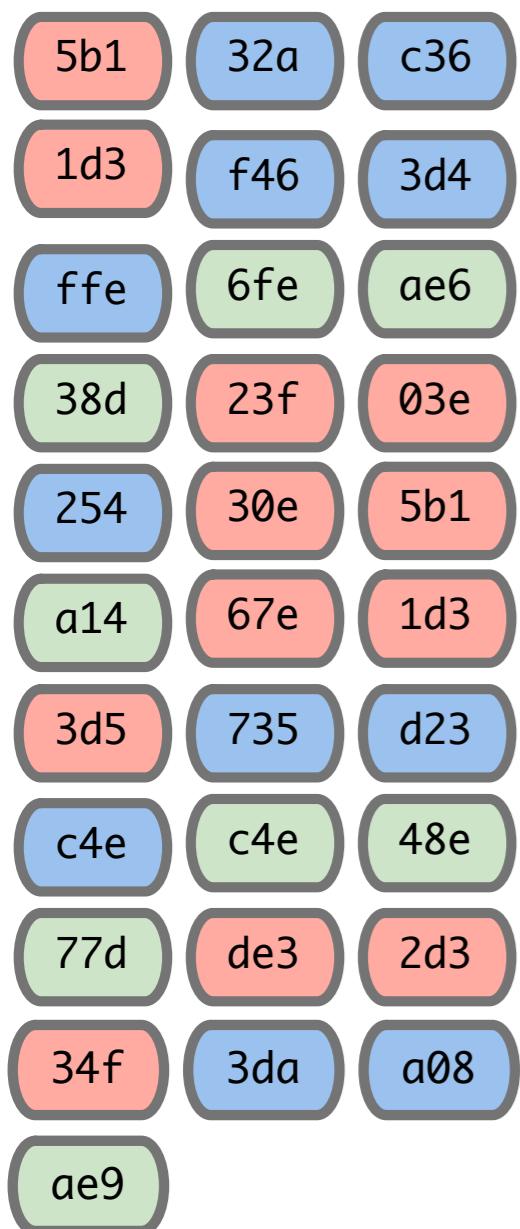


Working Directory



git add

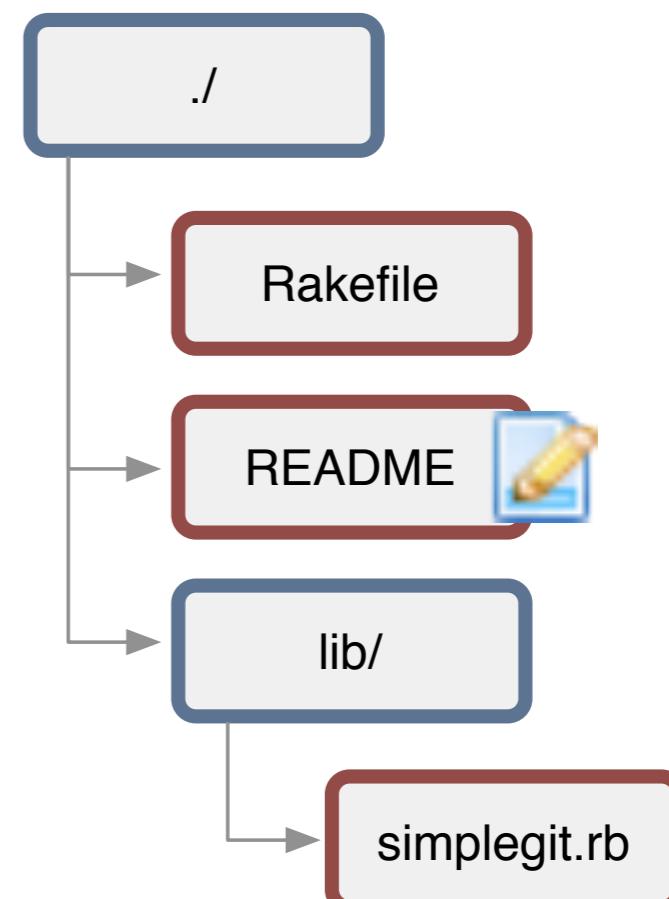
Repository



Index



Working Directory

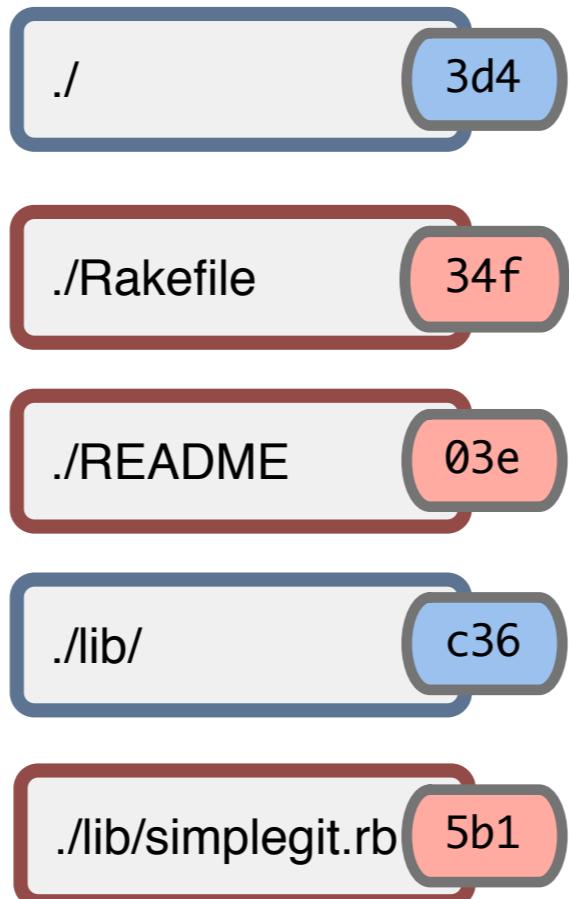


git commit

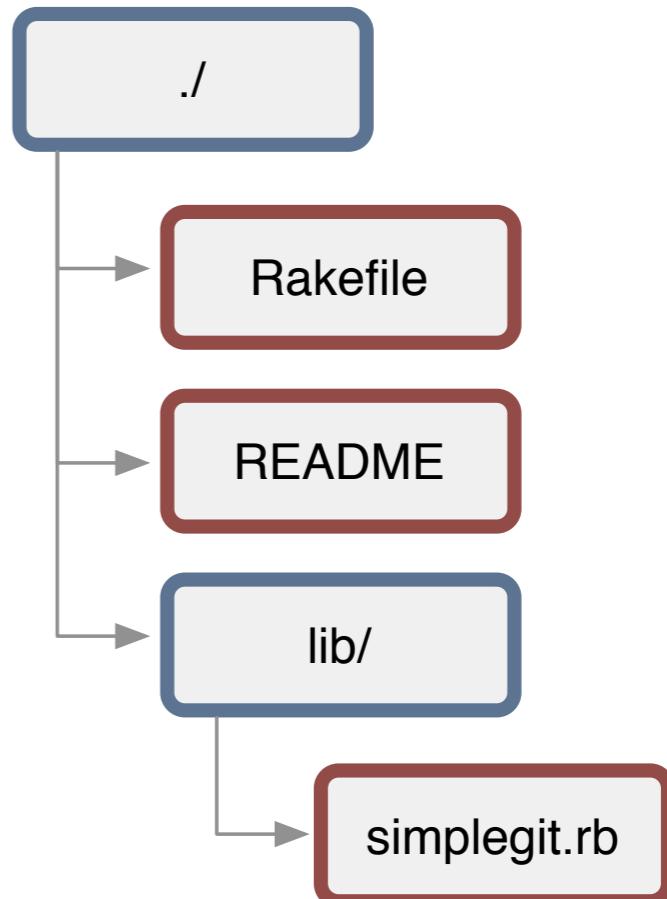
Repository



Index



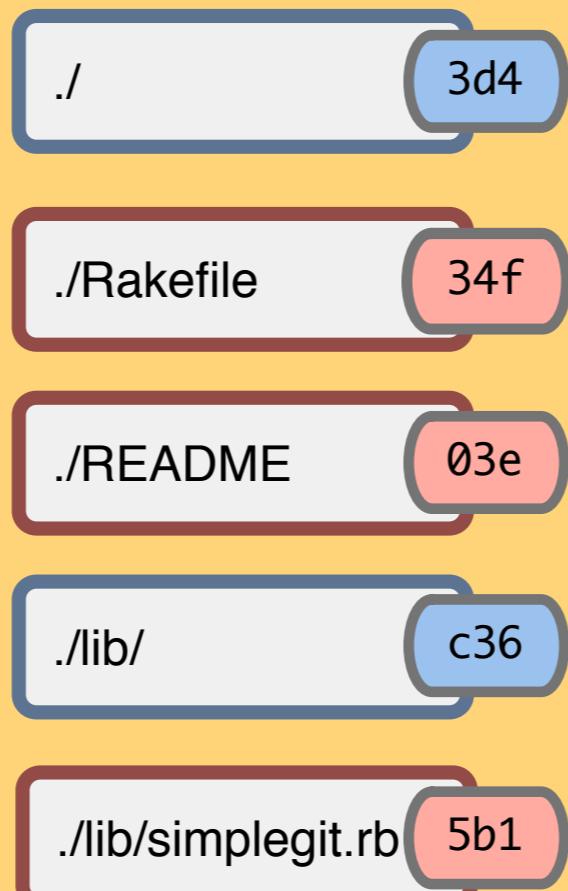
Working Directory



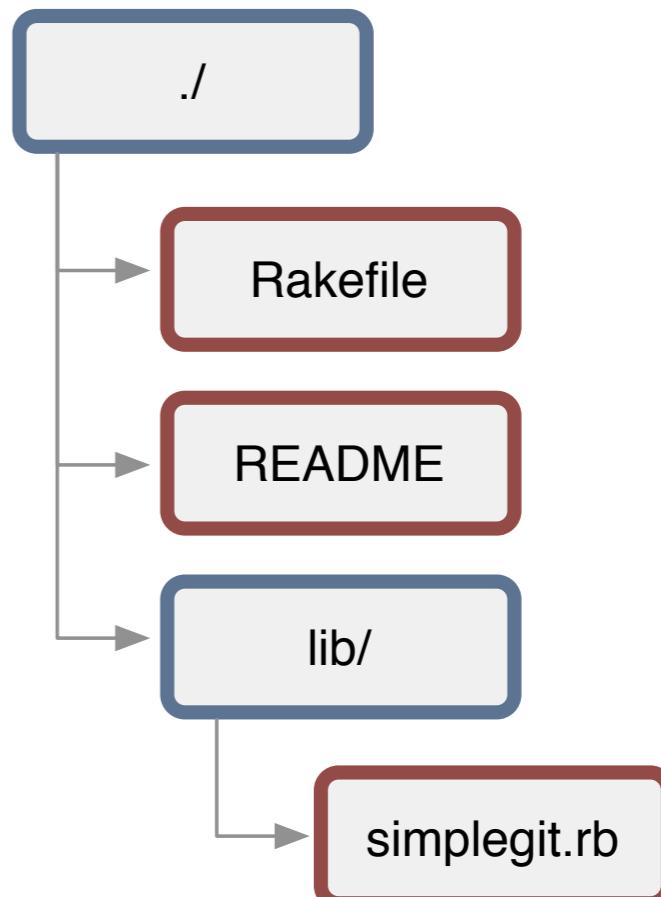
Repository



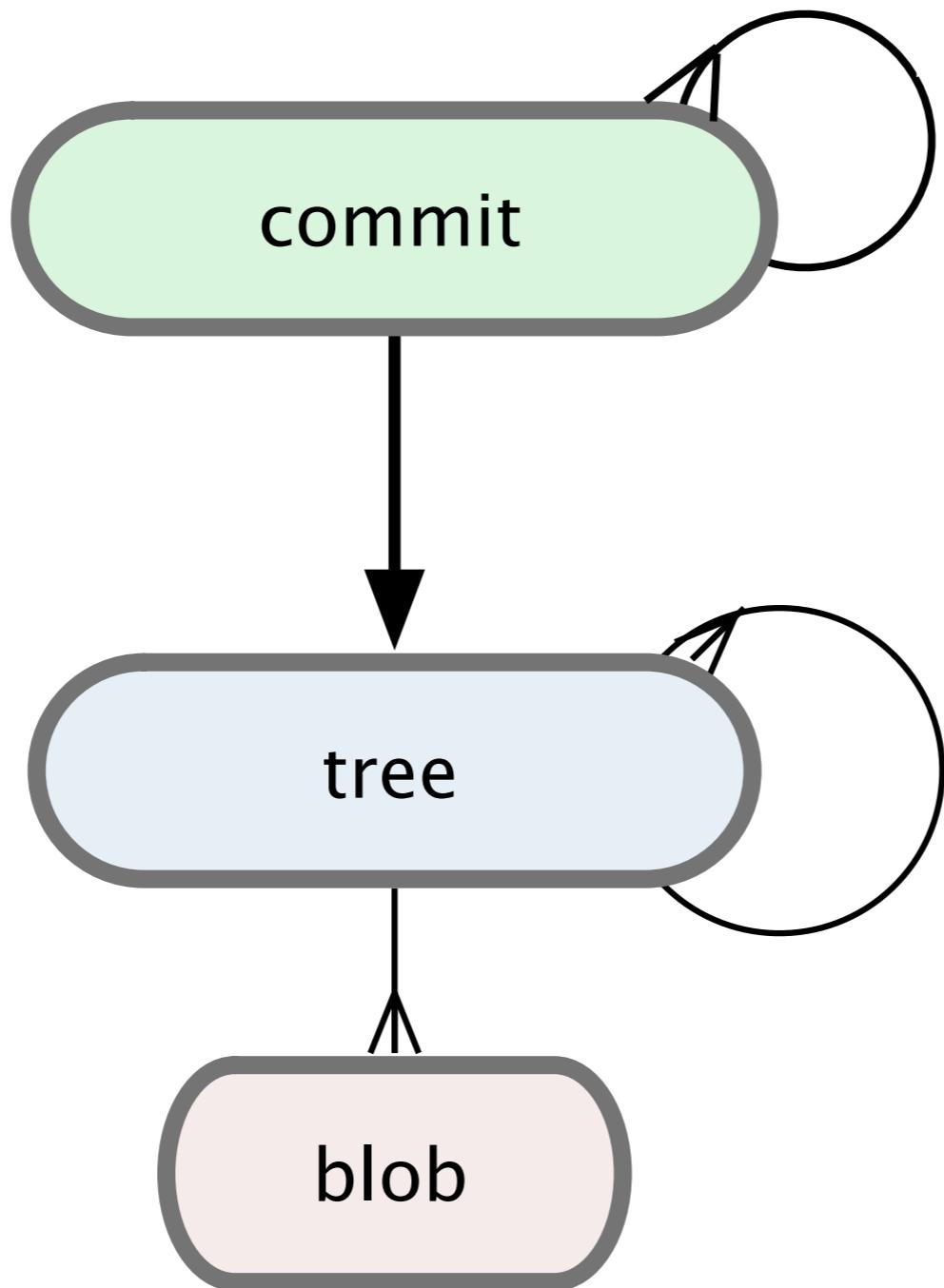
Index



Working Directory



object model

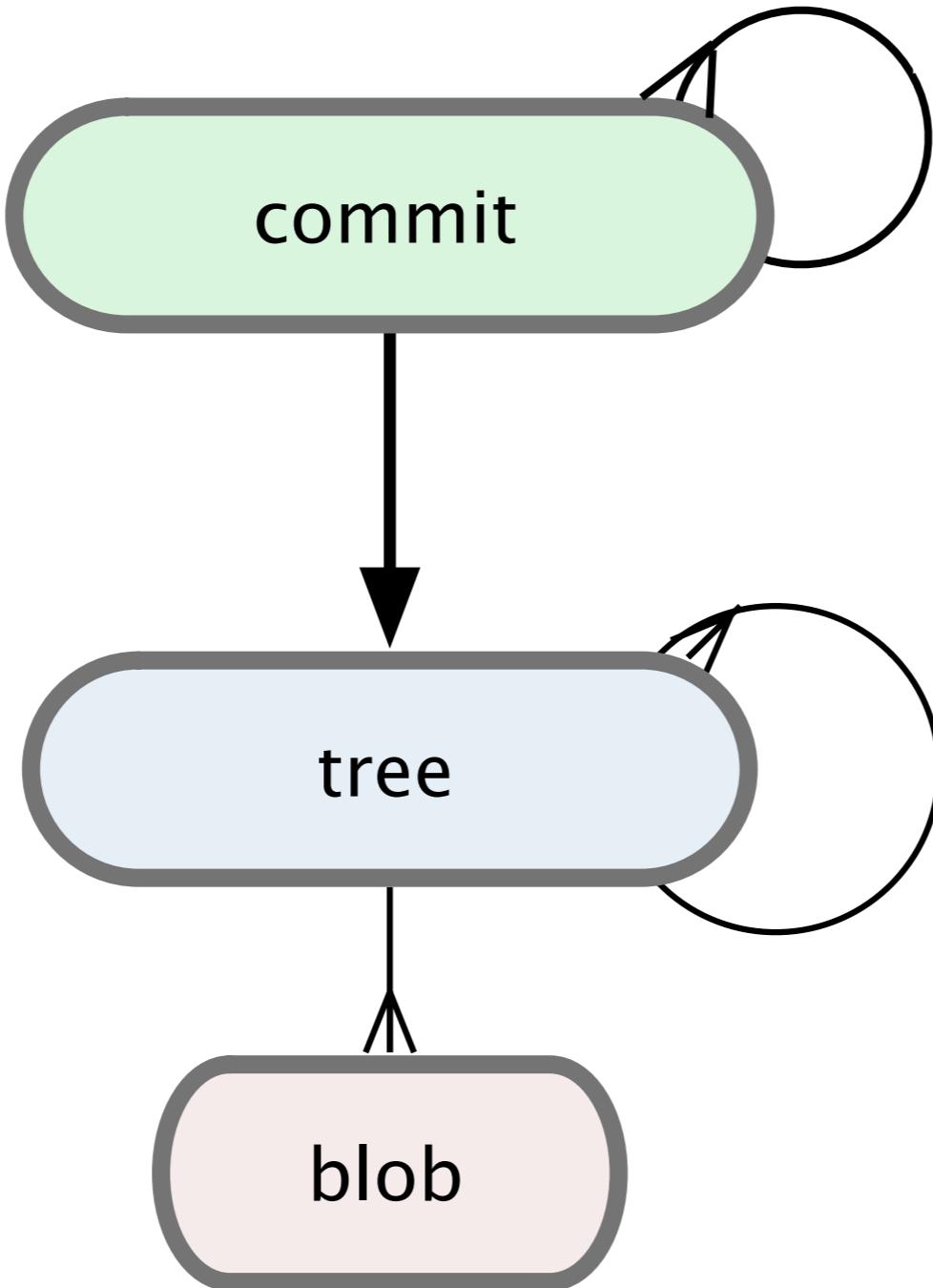


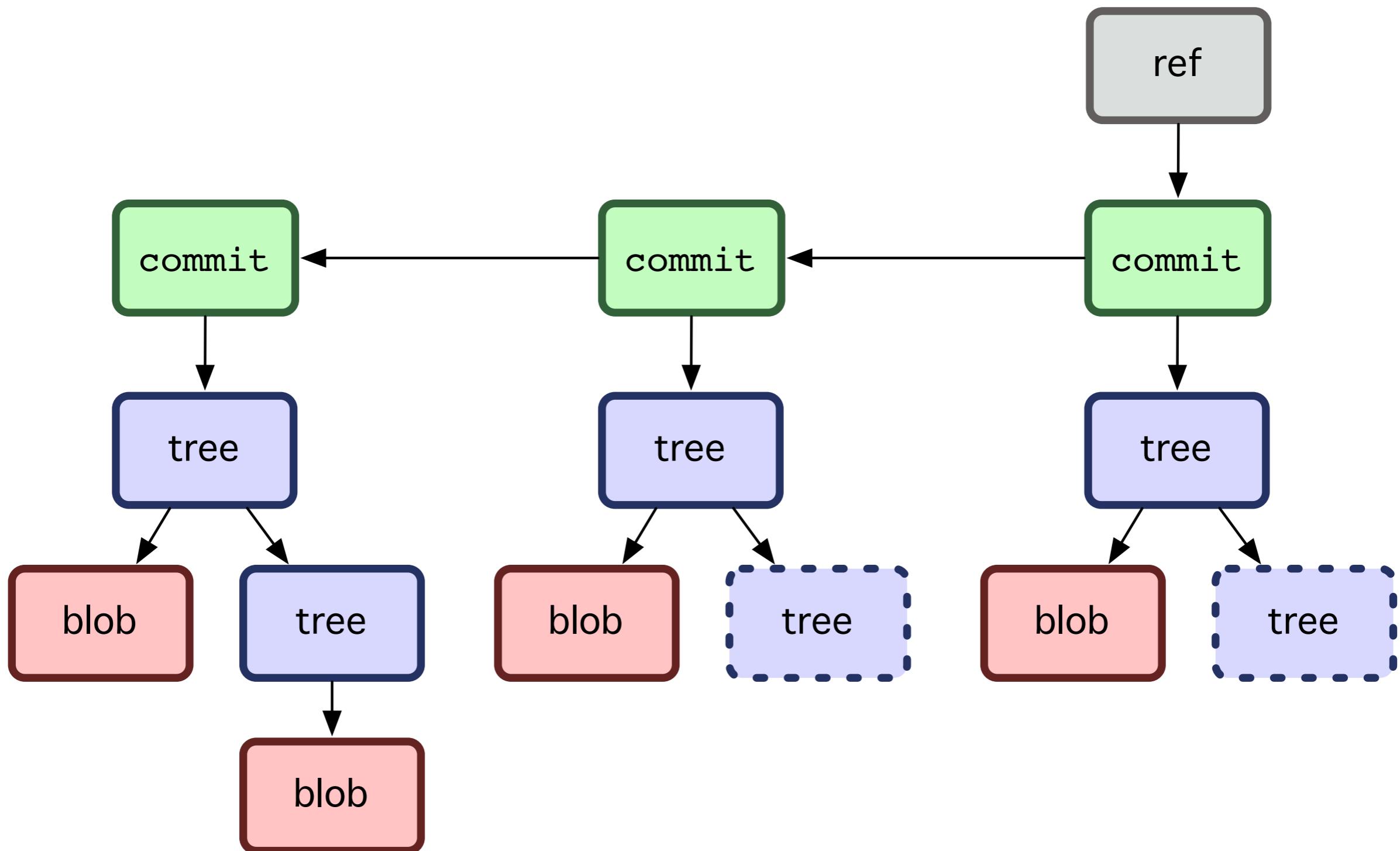
object model

pointer to a
snapshot

directory list

file contents



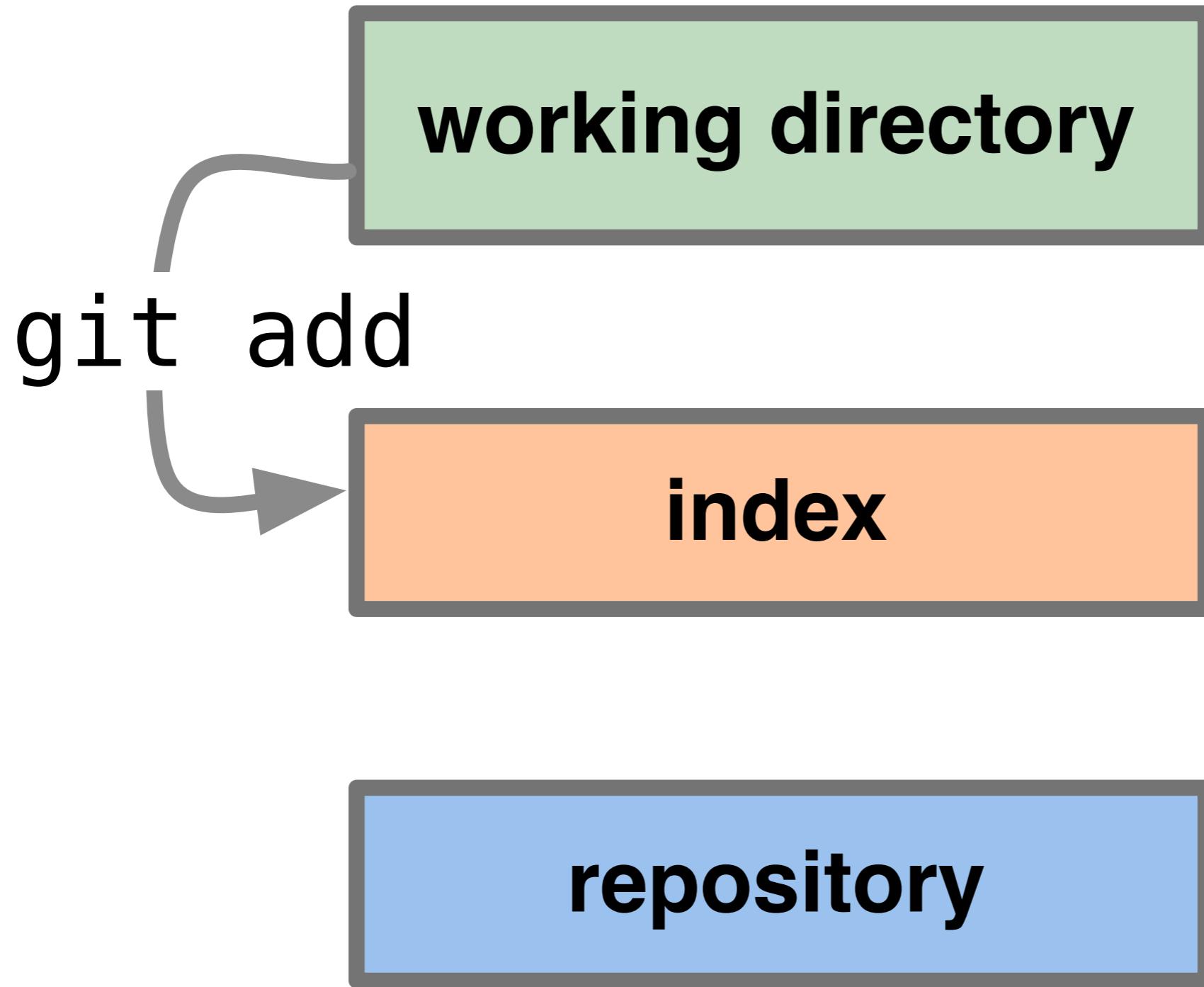


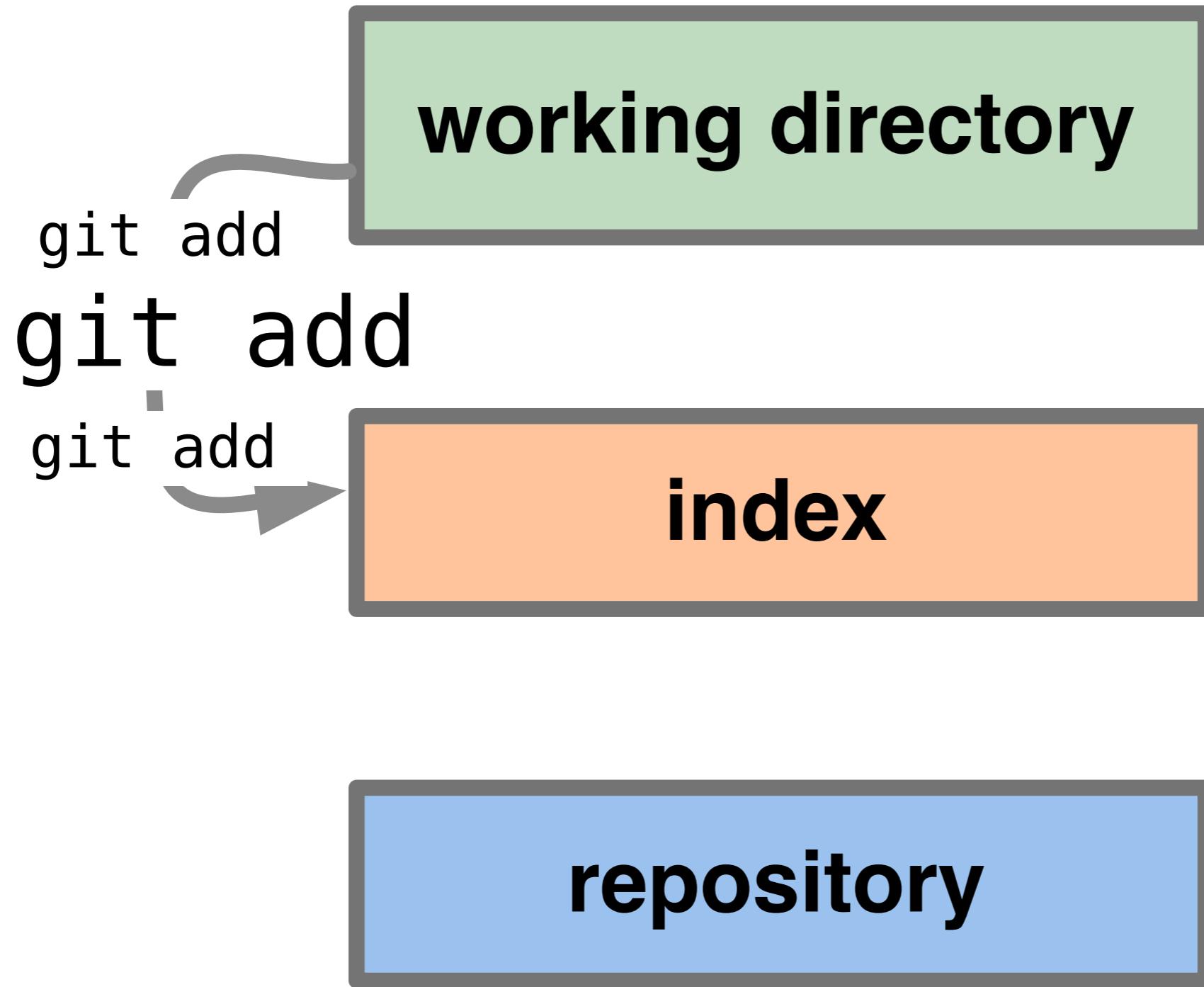
Bulk Staging

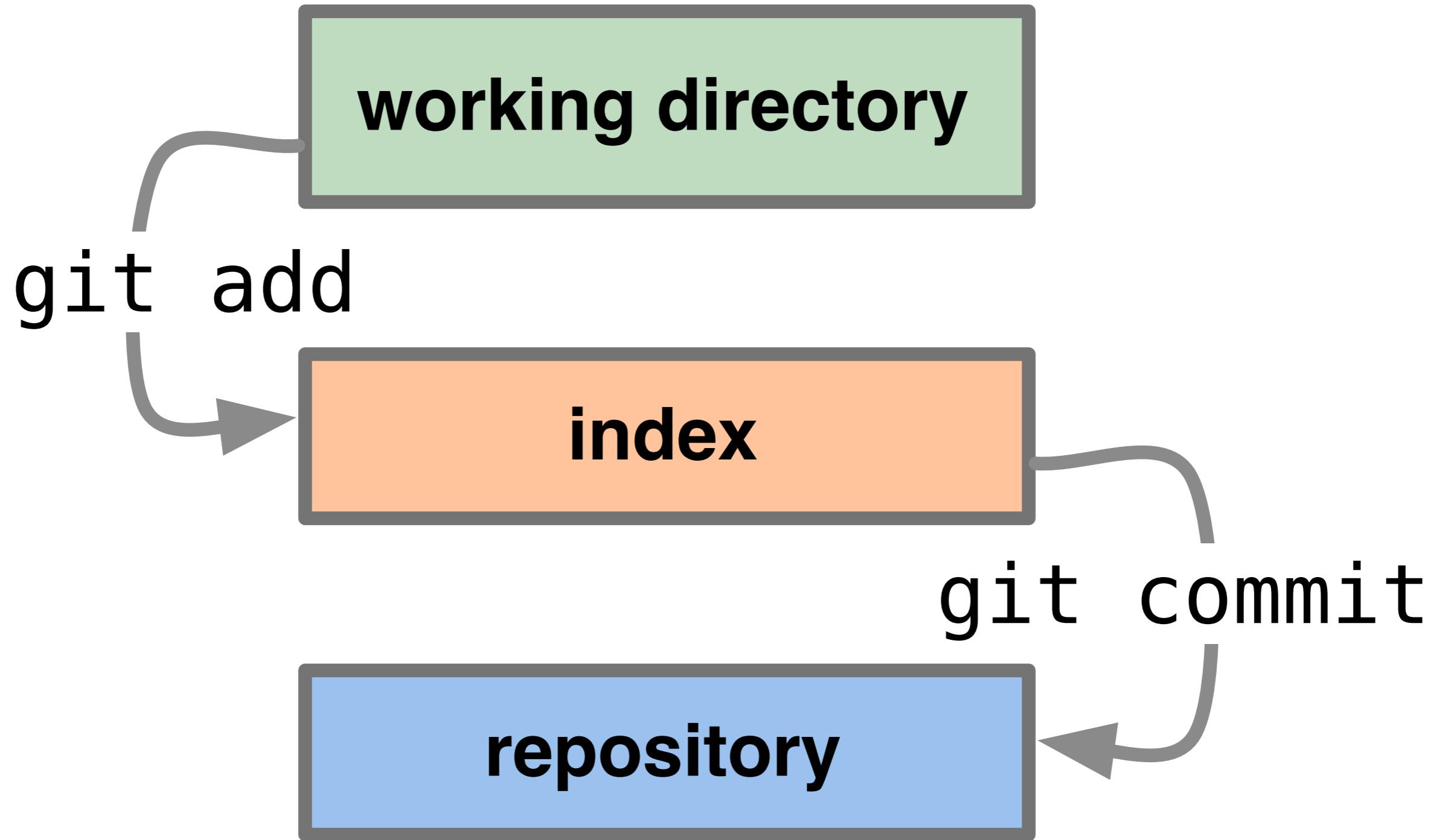
working directory

index

repository

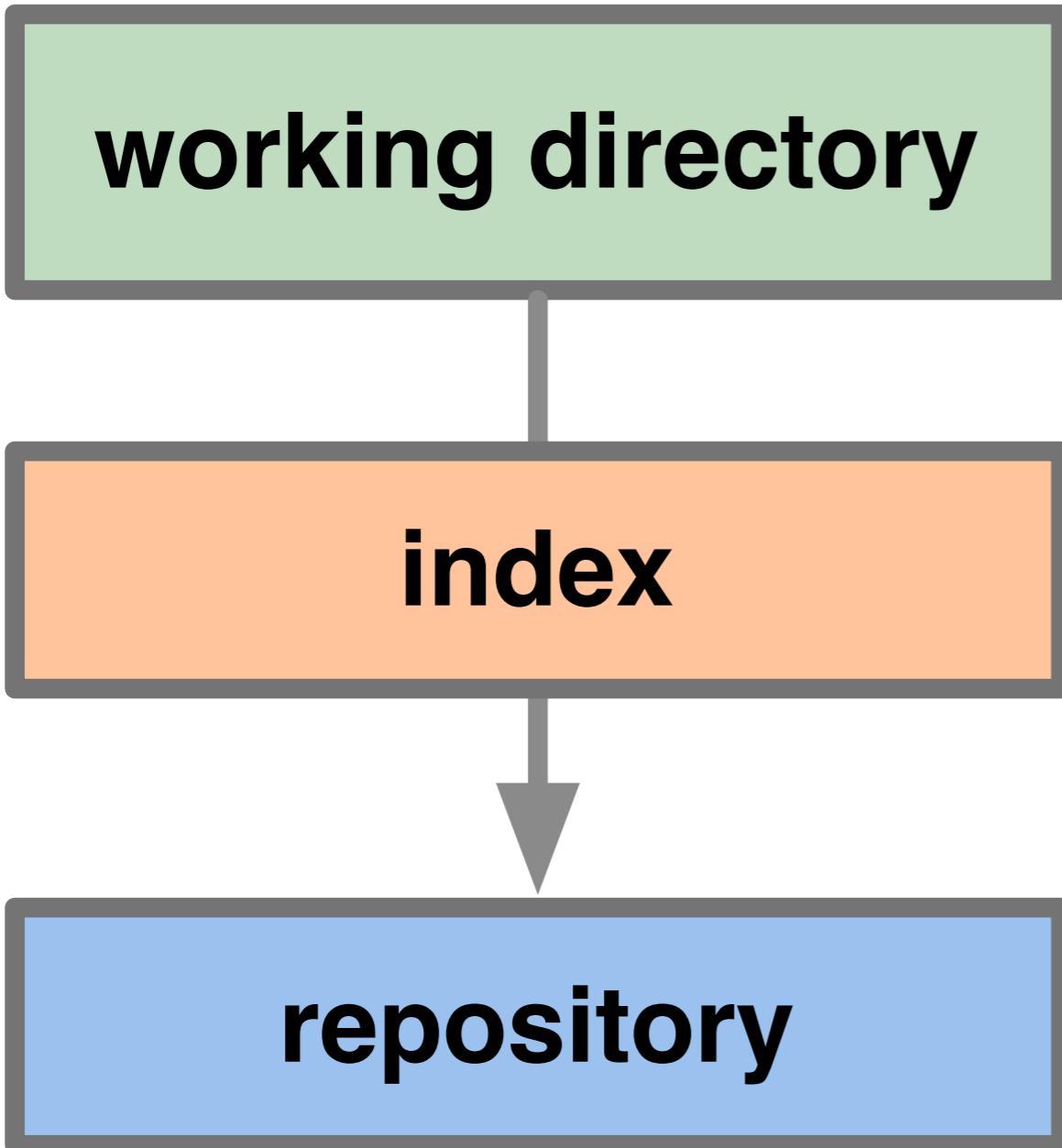






git add -u

git commit -a



git commit -a

git commit -a

git add -u
git commit

Patch Staging

git add -p

```
$ git status
```

```
$ git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes)
#
# modified:    ticgit.gemspec
#
no changes added to commit (use "git add" and/
```

```
$ git add -p
```

```
$ git add -p  
diff --git a/ticgit.gemspec b/ticgit.gemspec  
index 9c32bd4..a44667b 100644  
--- a/ticgit.gemspec  
+++ b/ticgit.gemspec  
@@ -1,7 +1,7 @@
```

```
Gem::Specification.new do |s|  
  s.platform = Gem::Platform::RUBY  
  s.name = "ticgit"  
-  s.version = "0.3.5"  
+  s.version = "0.3.6"  
  s.date = "2008-05-10"  
  s.author = "Scott Chacon"  
  s.email = "schacon@gmail.com"
```

```
Stage this hunk [y,n,a,d/,j,J,g,e,?]? y
```

```
$ git add -p
diff --git a/ticgit.gemspec b/ticgit.gemspec
index 9c32bd4..a44667b 100644
--- a/ticgit.gemspec
+++ b/ticgit.gemspec
@@ -1,7 +1,7 @@
Gem::Specification.new do |s|
  s.platform = Gem::Platform::RUBY
  s.name = "ticgit"
-  s.version = "0.3.5"
+  s.version = "0.3.6"
  s.date = "2008-05-10"
  s.author = "Scott Chacon"
  s.email = "schacon@gmail.com"
```

Stage this hunk [y,n,a,d/,j,J,g,e,?]?

```
$ git add -p  
diff --git a/ticgit.gemspec b/ticgit.gemspec  
index 9c32bd4..a44667b 100644  
--- a/ticgit.gemspec  
+++ b/ticgit.gemspec  
@@ -1,7 +1,7 @@
```

```
Gem::Specification.new do |s|  
  s.platform = Gem::Platform::RUBY  
  s.name = "ticgit"  
-  s.version = "0.3.5"  
+  s.version = "0.3.6"  
  s.date = "2008-05-10"  
  s.author = "Scott Chacon"  
  s.email = "schacon@gmail.com"
```

```
Stage this hunk [y,n,a,d/,j,J,g,e,?]?
```

y

```
@@ -9,9 +9,10 @@ Gem::Specification.new do |s|
  s.files      = ["lib/ticgit/base.rb", "lib/ticgit",
"lib/ticgit/comment.rb", "lib/ticgit/ticket.rb", "lib/
ticgit.rb", "bin/ti", "bin/ticgitweb"]

  s.bindir = 'bin'
-
-  s.executables << "ti"
-  s.executables << "ticgitweb"
+
+  s.executables = ["ti", "ticgitweb"]
+
+  s.default_executable = %q{ti}
  s.homepage = "http://github/schacon/ticgit"

  s.require_paths = ["lib", "bin"]
+
+  s.specification_version = 2 if
s.respond_to? :specification_version=
end
```

Stage this hunk [y,n,a,d,/ ,K,g,s,e,?]?

```
@@ -9,9 +9,10 @@ Gem::Specification.new do |s|
  s.files      = ["lib/ticgit/base.rb", "lib/ticgit",
"lib/ticgit/comment.rb", "lib/ticgit/ticket.rb", "lib/
ticgit.rb", "bin(ti", "bin/ticgitweb"]

  s.bindir = 'bin'
-
-  s.executables << "ti"
-  s.executables << "ticgitweb"
+
+  s.executables = ["ti", "ticgitweb"]
+
+  s.default_executable = %q{ti}
  s.homepage = "http://github/schacon/ticgit"

  s.require_paths = ["lib", "bin"]
+
+  s.specification_version = 2 if
s.respond_to? :specification_version=
end
```

Stage this hunk [y,n,a,d/,K,g,s,e,?]?

```
@@ -9,9 +9,10 @@ Gem::Specification.new do |s|
  s.files      = ["lib/ticgit/base.rb", "lib/ticgit",
"lib/ticgit/comment.rb", "lib/ticgit/ticket.rb", "lib/
ticgit.rb", "bin(ti", "bin/ticgitweb"]

  s.bindir = 'bin'
-
-  s.executables << "ti"
-  s.executables << "ticgitweb"
+
+  s.executables = ["ti", "ticgitweb"]
+  s.default_executable = %q{ti}
  s.homepage = "http://github/schacon/ticgit"

  s.require_paths = ["lib", "bin"]
+
+  s.specification_version = 2 if
s.respond_to? :specification_version=
end
```

Stage this hunk [y,n,a,d/,K,g,s,e,?]?



```
$ git status
```

```
$ git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:    ticgit.gemspec
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes)
#
# modified:    ticgit.gemspec
#
```

```
$ git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:    ticgit.gemspec
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes)
#
# modified:    ticgit.gemspec
#
```



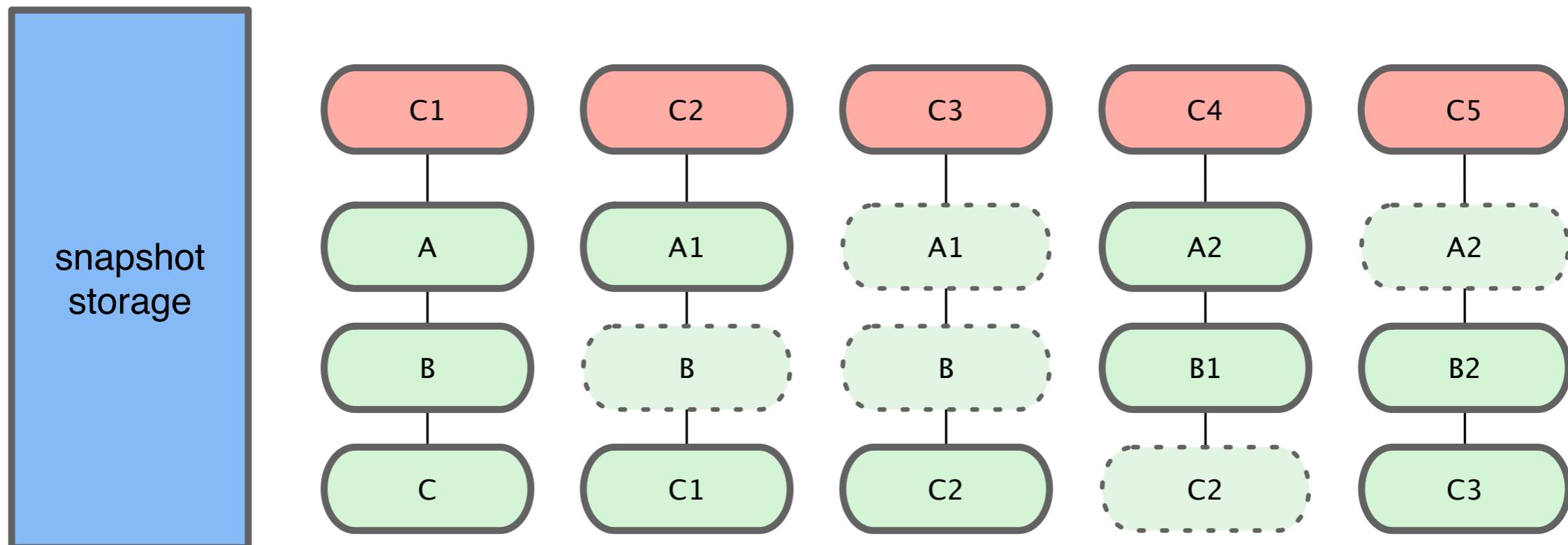
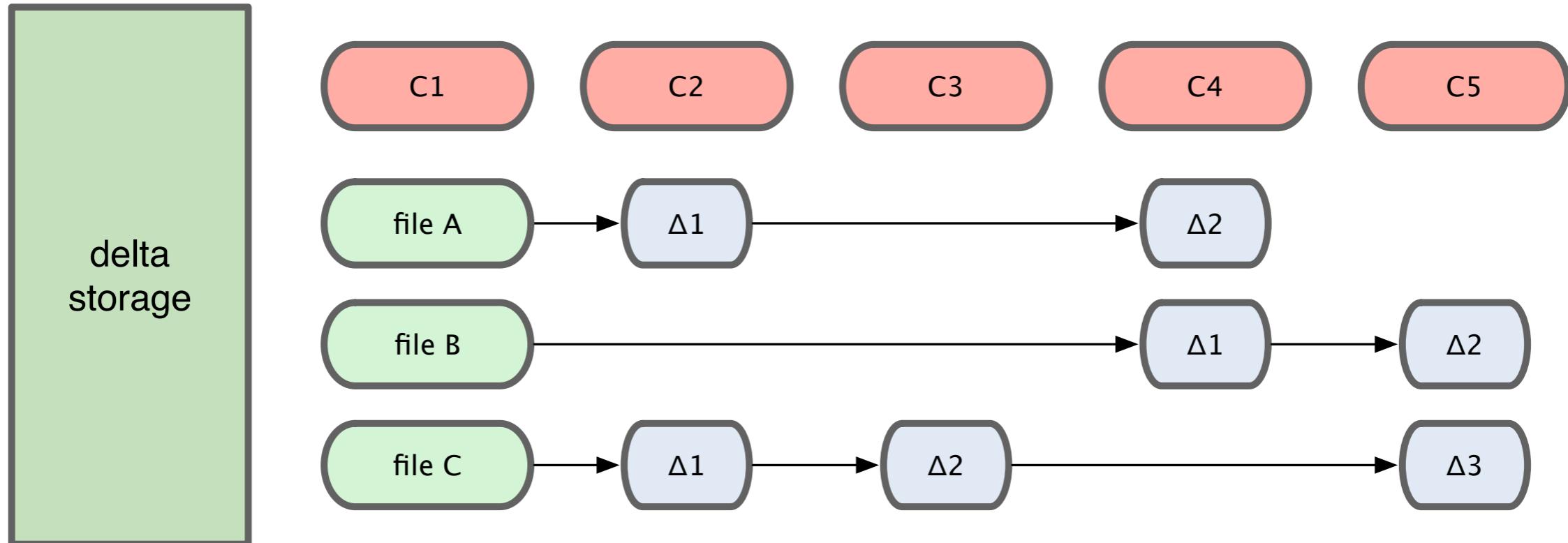
```
$ git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:    ticgit.gemspec
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be com-
#     (use "git checkout -- <file>..." to discard changes)
#
# modified:    ticgit.gemspec
```

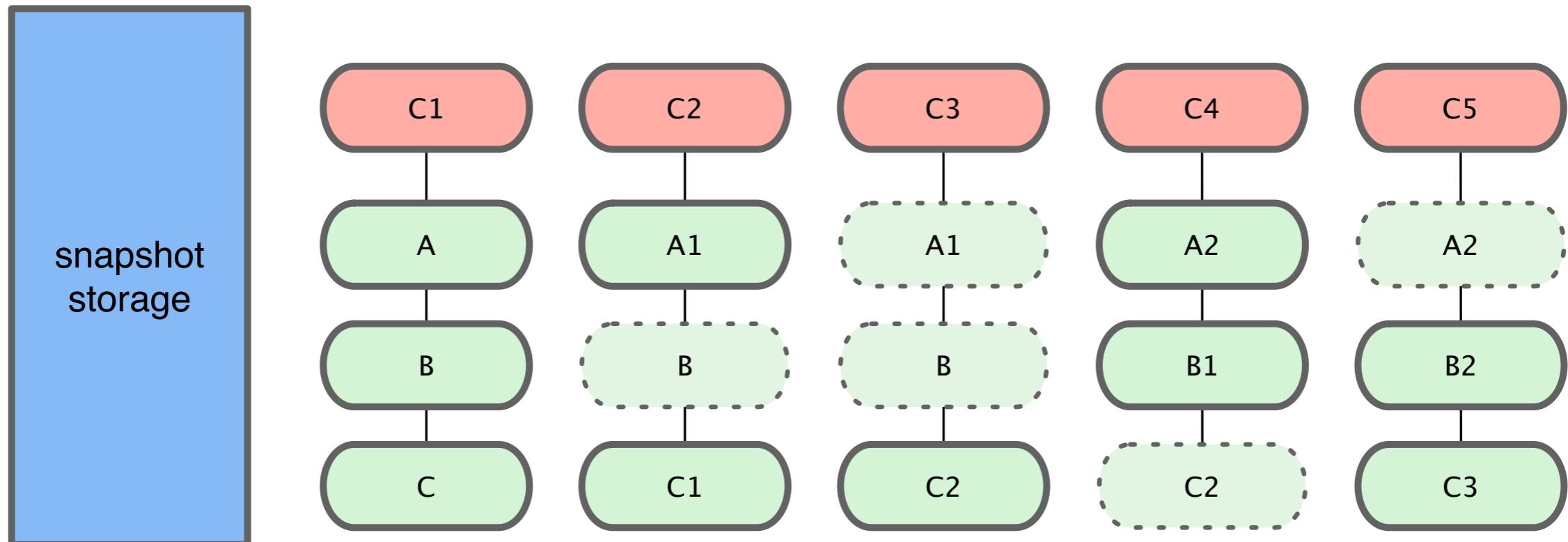
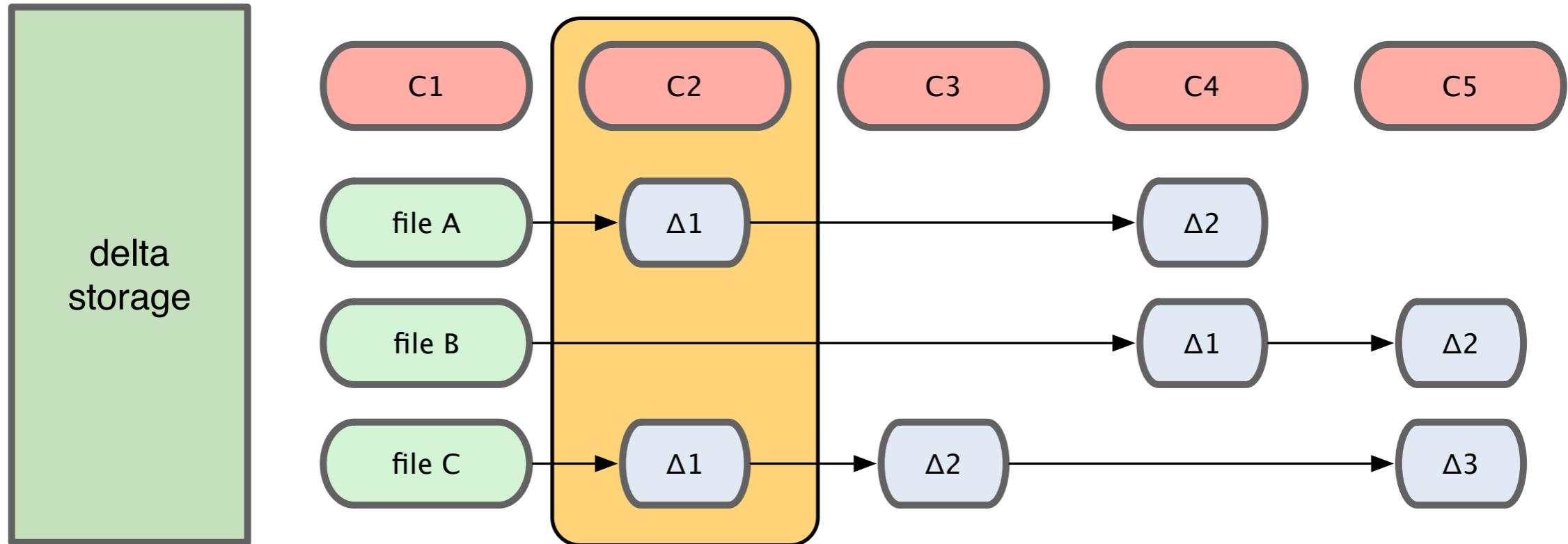


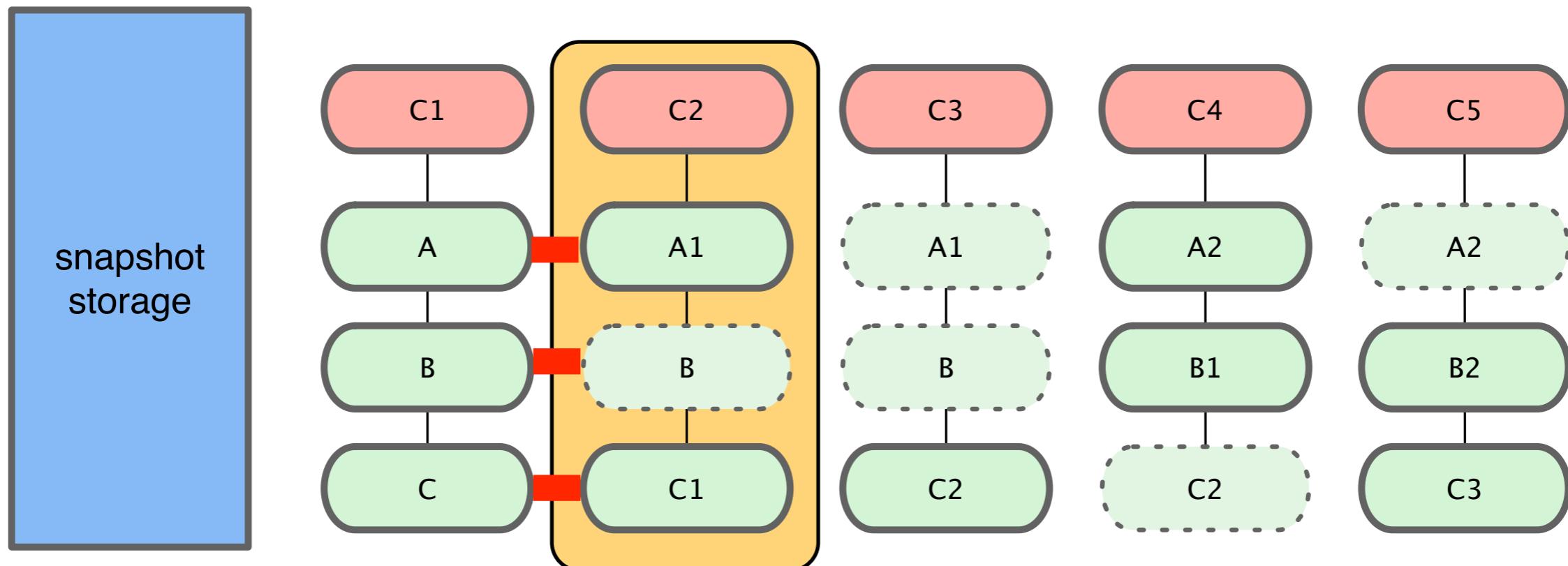
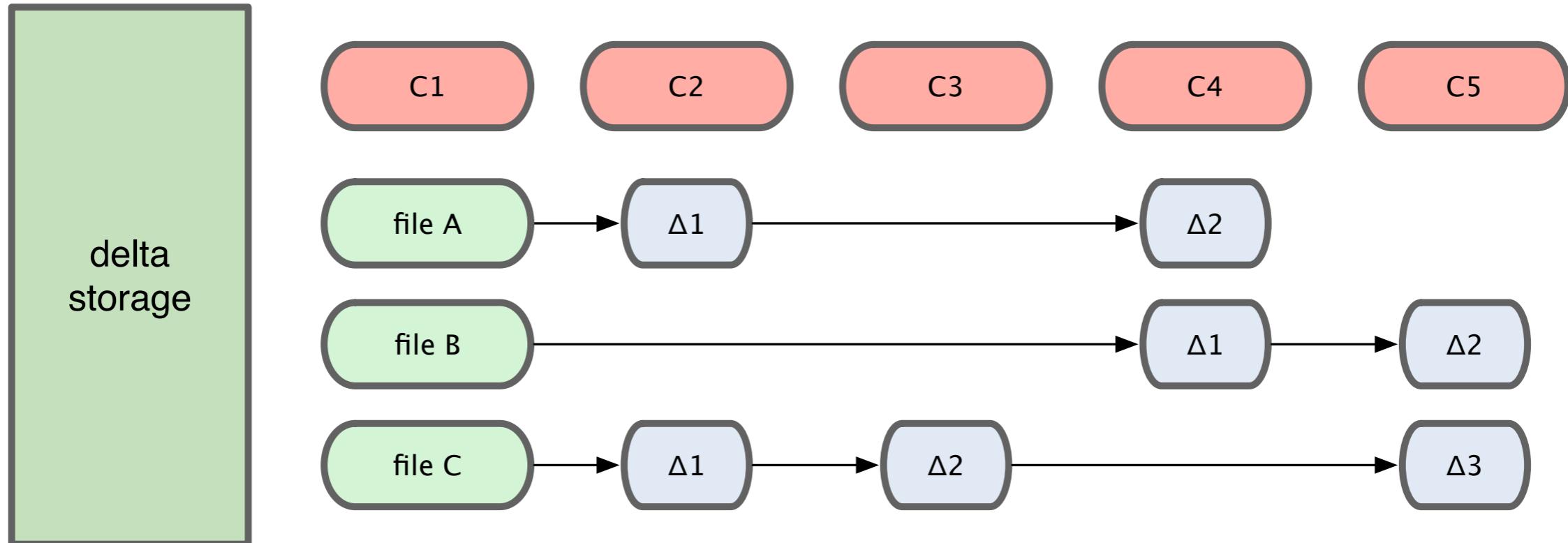
partially stage files

git add -p

Changes







git diff

```
diff --git a/main.py b/main.py
index 6db8b97..b9bcc62 100755
--- a/main.py
+++ b/main.py
@@ -2,10 +2,12 @@
 import wsgiref.handlers
 from google.appengine.ext import webapp

+# this program prints out 'hello world'
+
class MainHandler(webapp.RequestHandler):

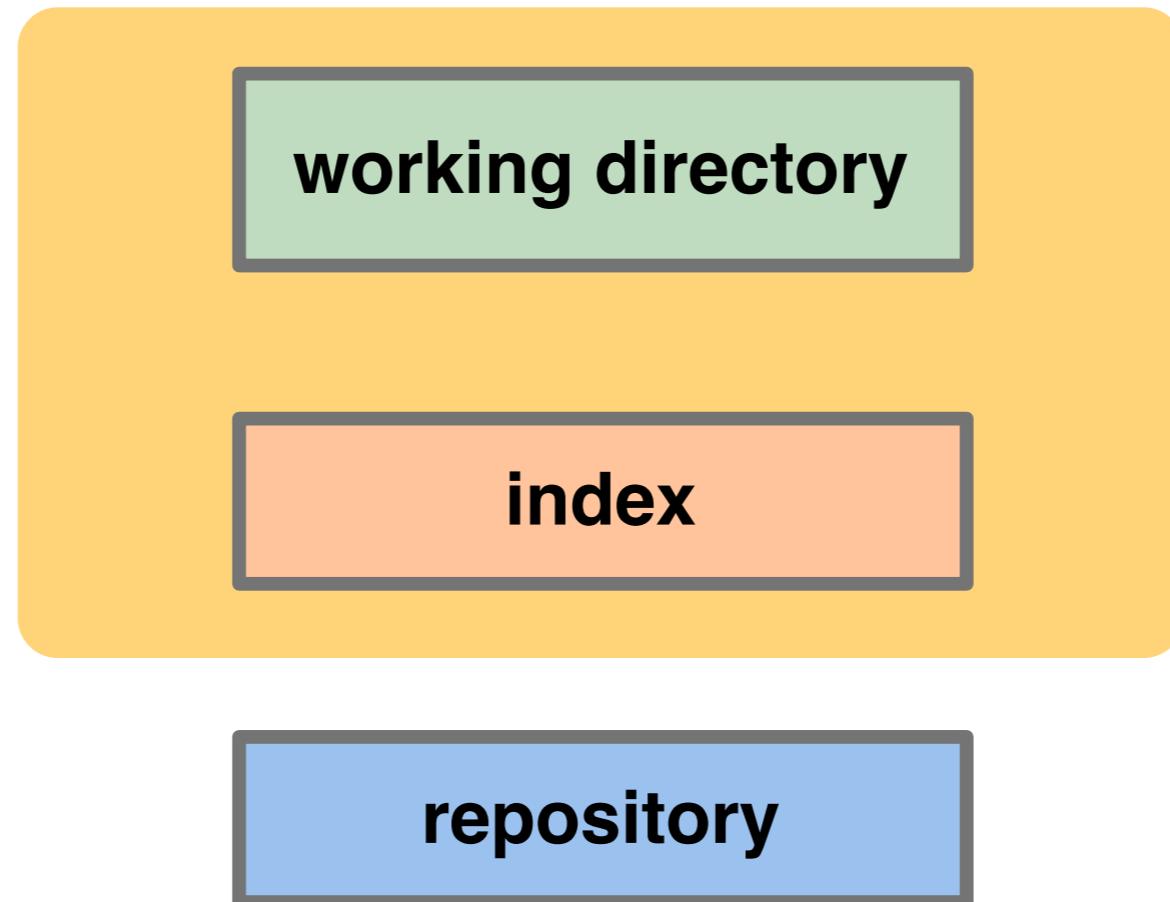
    def get(self):
-        self.response.out.write('Hello world!')
+        self.response.out.write('Hola mundo!')

def main():
    application = webapp.WSGIApplication([( '/', MainHandler )],
```

What is not yet staged?

git diff

git diff



What is staged?

git diff --cached

`git diff --cached`

working directory

index

repository

git diff --stat

```
$ git diff --stat
kidgloves.rb |      2 +-  
1 files changed, 1 insertions(+),  
1 deletions(-)
```

Unified Diff

git diff > change.patch

`git diff > change.patch`

`patch -p1 < change.patch`

git diff > change.patch

patch -p1 < change.patch

or

git apply change.patch

History

git log

```
$>git log
```

```
$>git log
```

```
commit 310154e3c7db47d8bac935c2c43aee6afac11aae
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 13 10:49:15 2008 -0700
```

updated README formatting and added blame

```
commit f7f3f6dd8fd3fa40f052427c32785a0fa01aaa5f
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:45:01 2008 -0700
```

changed my name a bit

```
commit 710f0f8d2cdf5af87033b9ec08859a505f9a6af5
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:34:16 2008 -0700
```

added ls-files

```
commit c110d7ff8cfb86fd5cce9a8aee462678dbb4ef9b
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 6 12:13:36 2008 -0700
```

made the ls-tree function recursive and list trees

```
$>git log
```

```
commit 310154e3c7db47d8bac935c2c43aee6afac11aae
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 13 10:49:15 2008 -0700
```

```
updated README formatting and added blame
```

```
commit f7f3f6dd8fd3fa40f052427c32785a0fa01aaa5f
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:45:01 2008 -0700
```

```
changed my name a bit
```

```
commit 710f0f8d2cdf5af87033b9ec08859a505f9a6af5
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:34:16 2008 -0700
```

```
added ls-files
```

```
commit c110d7ff8cfb86fd5cce9a8aee462678dbb4ef9b
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 6 12:13:36 2008 -0700
```

```
made the ls-tree function recursive and list trees
```

```
$>git log
```

```
commit 310154e3c7db47d8bac935c2c43aee6afac11aae
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 13 10:49:15 2008 -0700
```

```
updated README formatting and added blame
```

```
commit f7f3f6dd8fd3fa40f052427c32785a0fa01aaa5f
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:45:01 2008 -0700
```

```
changed my name a bit
```

```
commit 710f0f8d2cdf5af87033b9ec08859a505f9a6af5
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:34:16 2008 -0700
```

```
added ls-files
```

```
commit c110d7ff8cfb86fd5cce9a8aee462678dbb4ef9b
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 6 12:13:36 2008 -0700
```

```
made the ls-tree function recursive and list trees
```

```
$>git log
```

```
commit 310154e3c7db47d8bac935c2c43aee6afac11aae
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 13 10:49:15 2008 -0700
```

updated README formatting and added blame

```
commit f7f3f6dd8fd3fa40f052427c32785a0fa01aaa5f
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:45:01 2008 -0700
```

changed my name a bit

```
commit 710f0f8d2cdf5af87033b9ec08859a505f9a6af5
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:34:16 2008 -0700
```

added ls-files

```
commit c110d7ff8cfb86fd5cce9a8aee462678dbb4ef9b
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 6 12:13:36 2008 -0700
```

made the ls-tree function recursive and list trees

```
$>git log
```

```
commit 310154e3c7db47d8bac935c2c43aee6afac11aae
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 13 10:49:15 2008 -0700
```

updated README formatting and added blame

```
commit f7f3f6dd8fd3fa40f052427c32785a0fa01aaa5f
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:45:01 2008 -0700
```

changed my name a bit

```
commit 710f0f8d2cdf5af87033b9ec08859a505f9a6af5
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:34:16 2008 -0700
```

added ls-files

```
commit c110d7ff8cfb86fd5cce9a8aee462678dbb4ef9b
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 6 12:13:36 2008 -0700
```

made the ls-tree function recursive and list trees

```
$>git log
```

```
commit 310154e3c7db47d8bac935c2c43aee6afac11aae
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 13 10:49:15 2008 -0700
```

updated README formatting and added blame

```
commit f7f3f6dd8fd3fa40f052427c32785a0fa01aaa5f
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:45:01 2008 -0700
```

changed my name a bit

```
commit 710f0f8d2cdf5af87033b9ec08859a505f9a6af5
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:34:16 2008 -0700
```

added ls-files

```
commit c110d7ff8cfb86fd5cce9a8aee462678dbb4ef9b
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 6 12:13:36 2008 -0700
```

made the ls-tree function recursive and list trees

```
$>git log
```

```
commit 310154e3c7db47d8bac935c2c43aee6afac11aae
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 13 10:49:15 2008 -0700
```

updated README formatting and added blame

```
commit f7f3f6dd8fd3fa40f052427c32785a0fa01aaa5f
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:45:01 2008 -0700
```

changed my name a bit

```
commit 710f0f8d2cdf5af87033b9ec08859a505f9a6af5
```

```
Author: Magnus Chacon <mchacon@gmail.com>
```

```
Date: Sun Apr 13 10:34:16 2008 -0700
```

added ls-files

```
commit c110d7ff8cfb86fd5cce9a8aee462678dbb4ef9b
```

```
Author: Scott Chacon <schacon@gmail.com>
```

```
Date: Sun Apr 6 12:13:36 2008 -0700
```

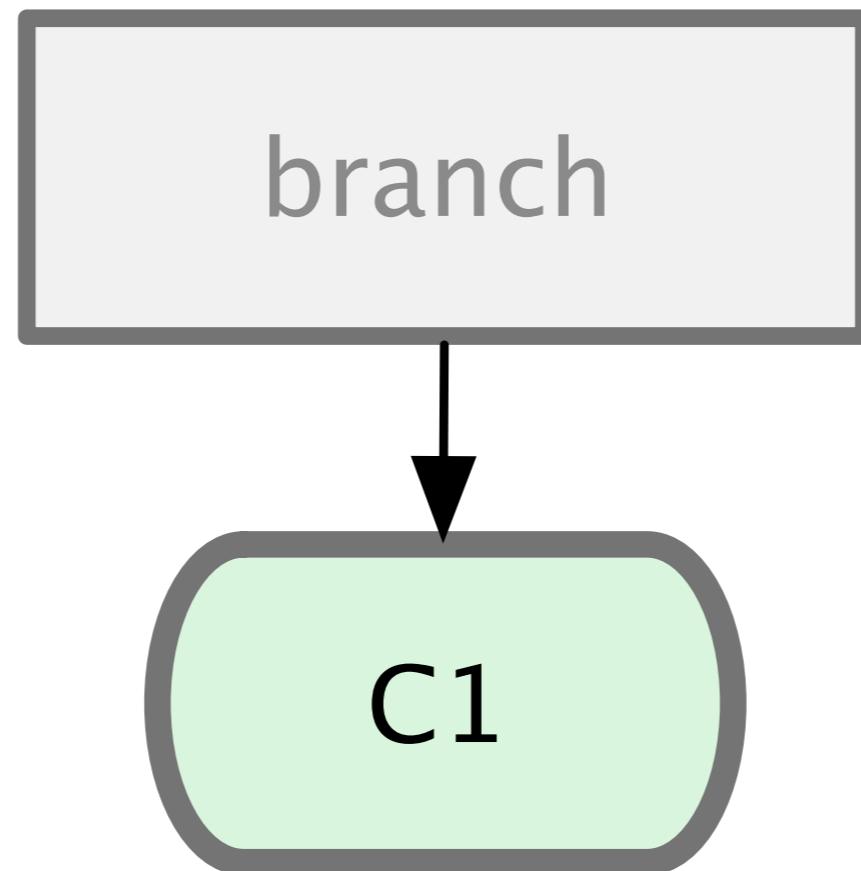
made the ls-tree function recursive and list trees

Branching and Merging

branches

branches

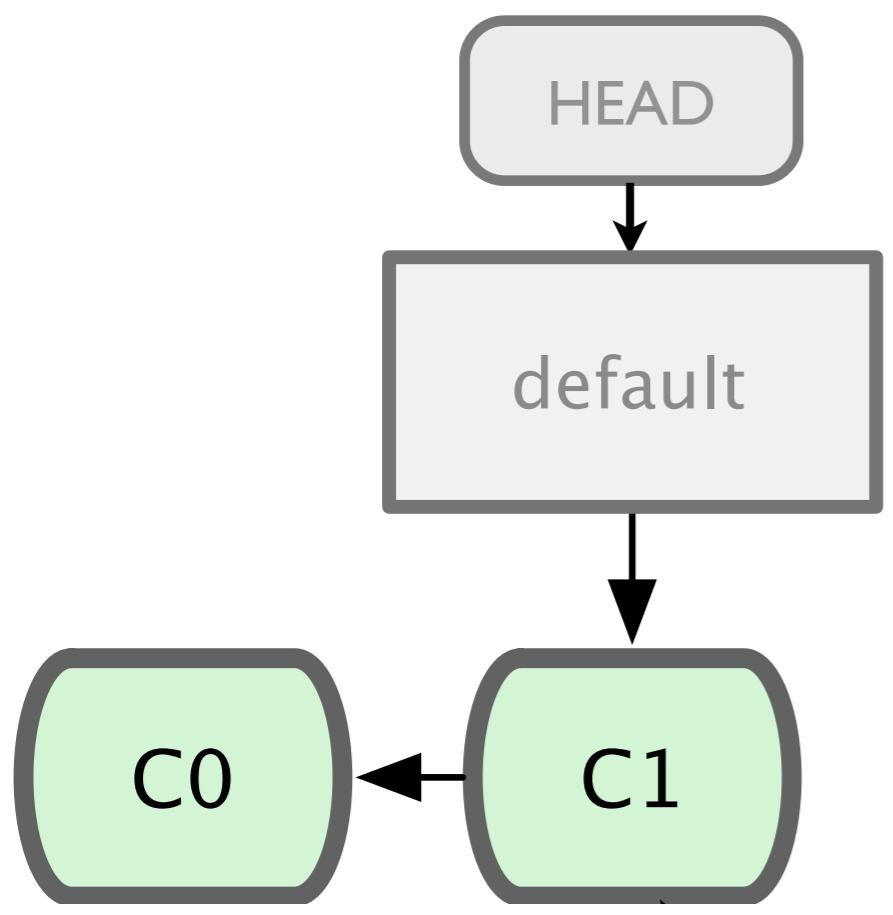
lightweight, movable
pointers to a commit

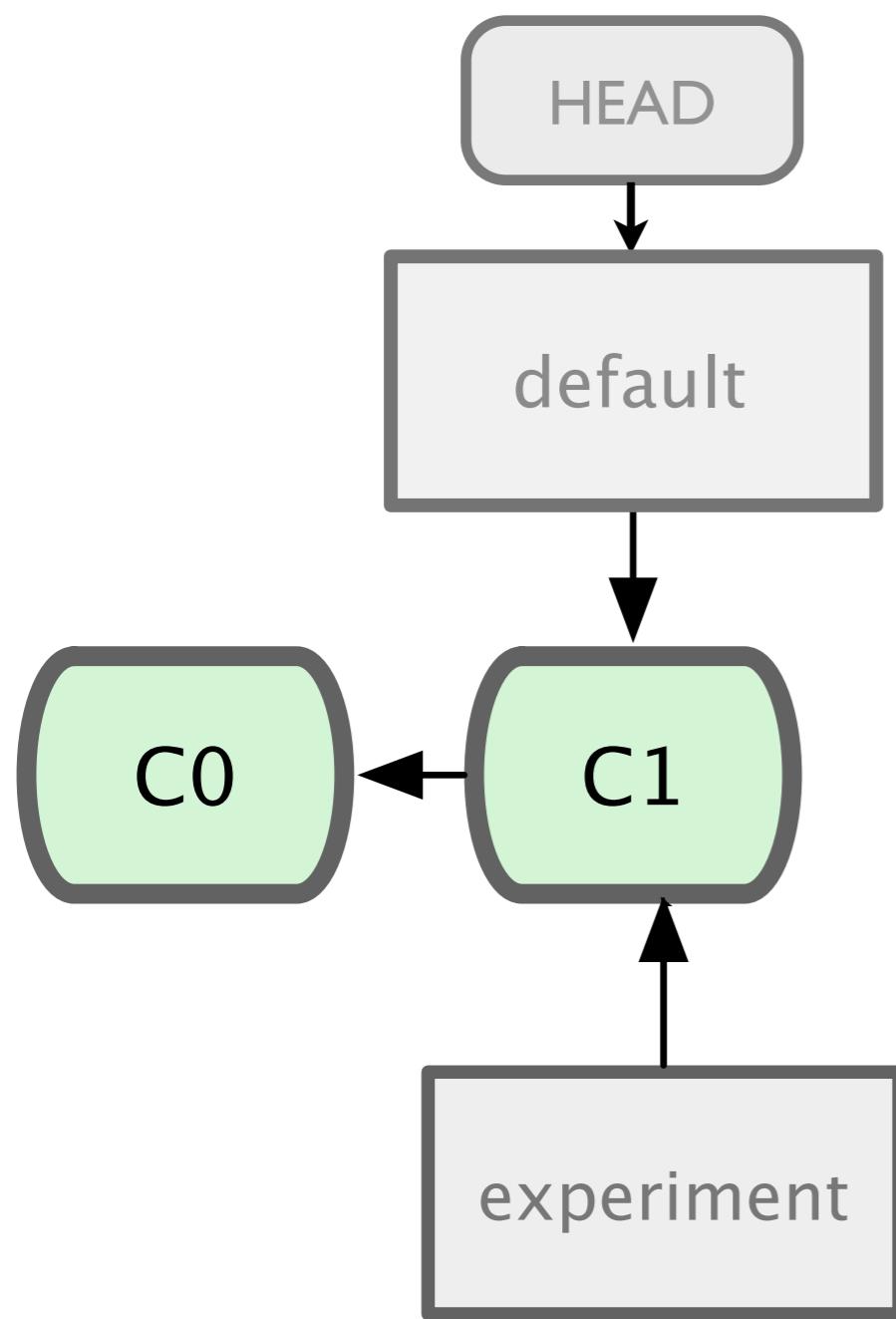


branching

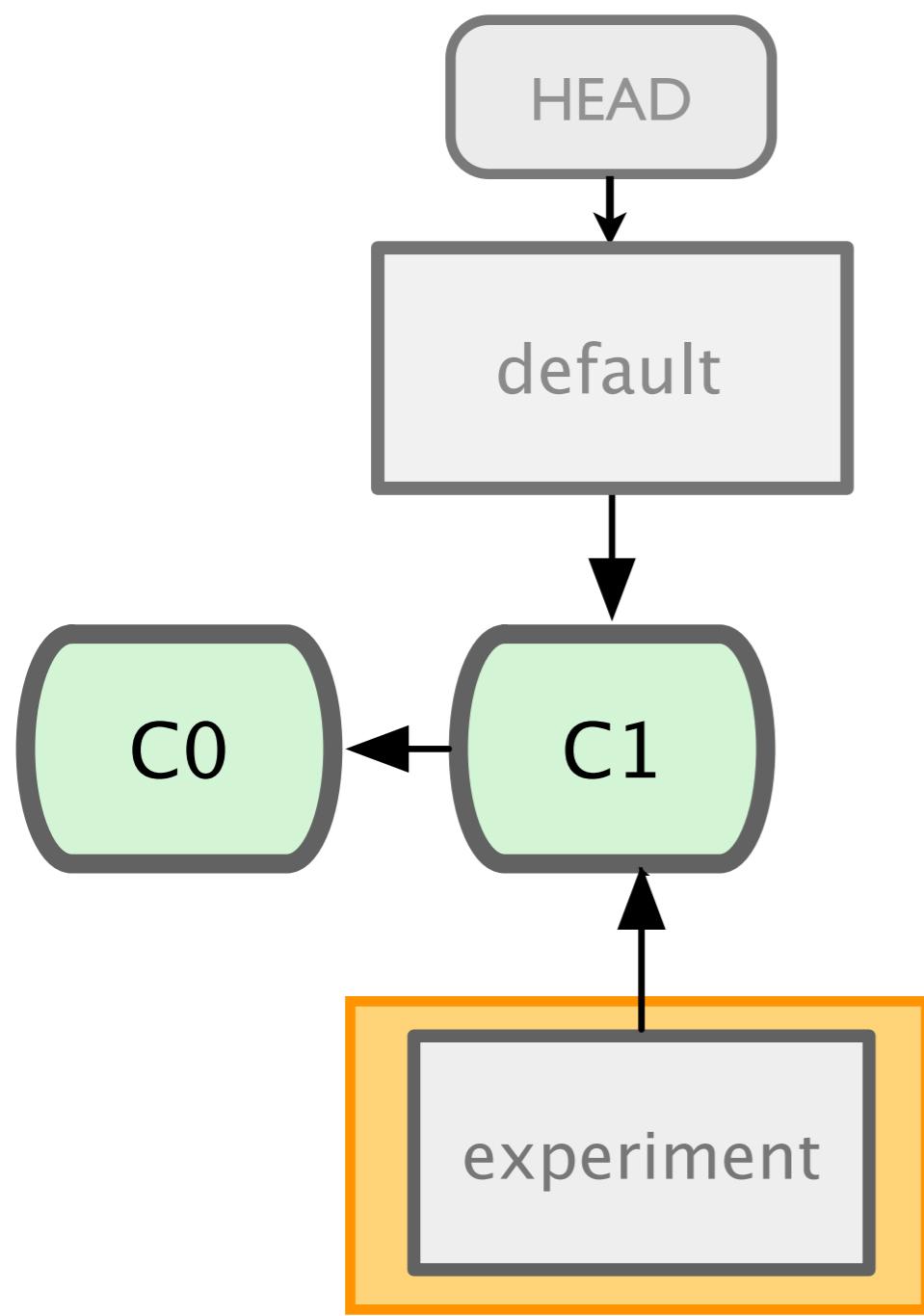
git branch

git checkout

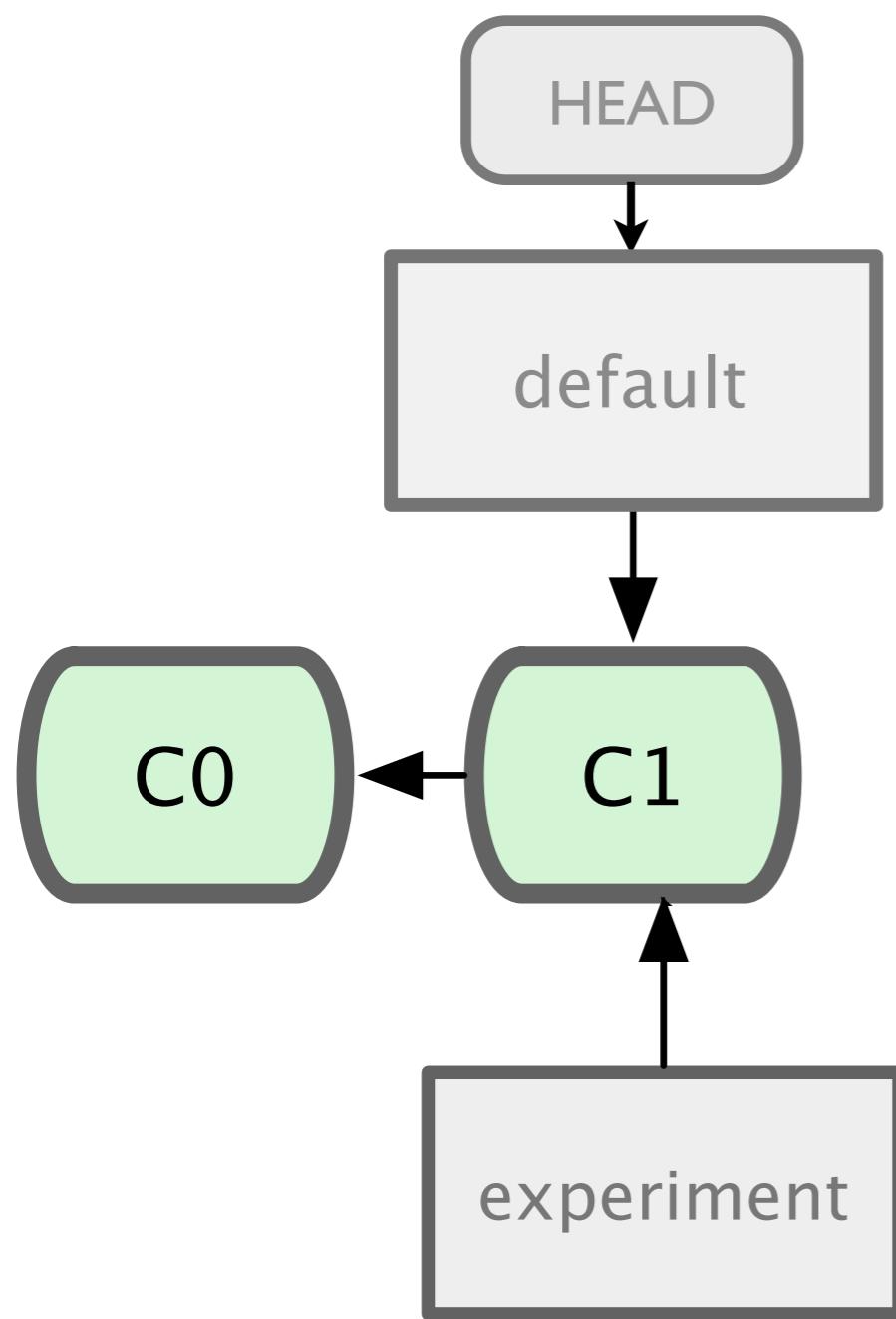




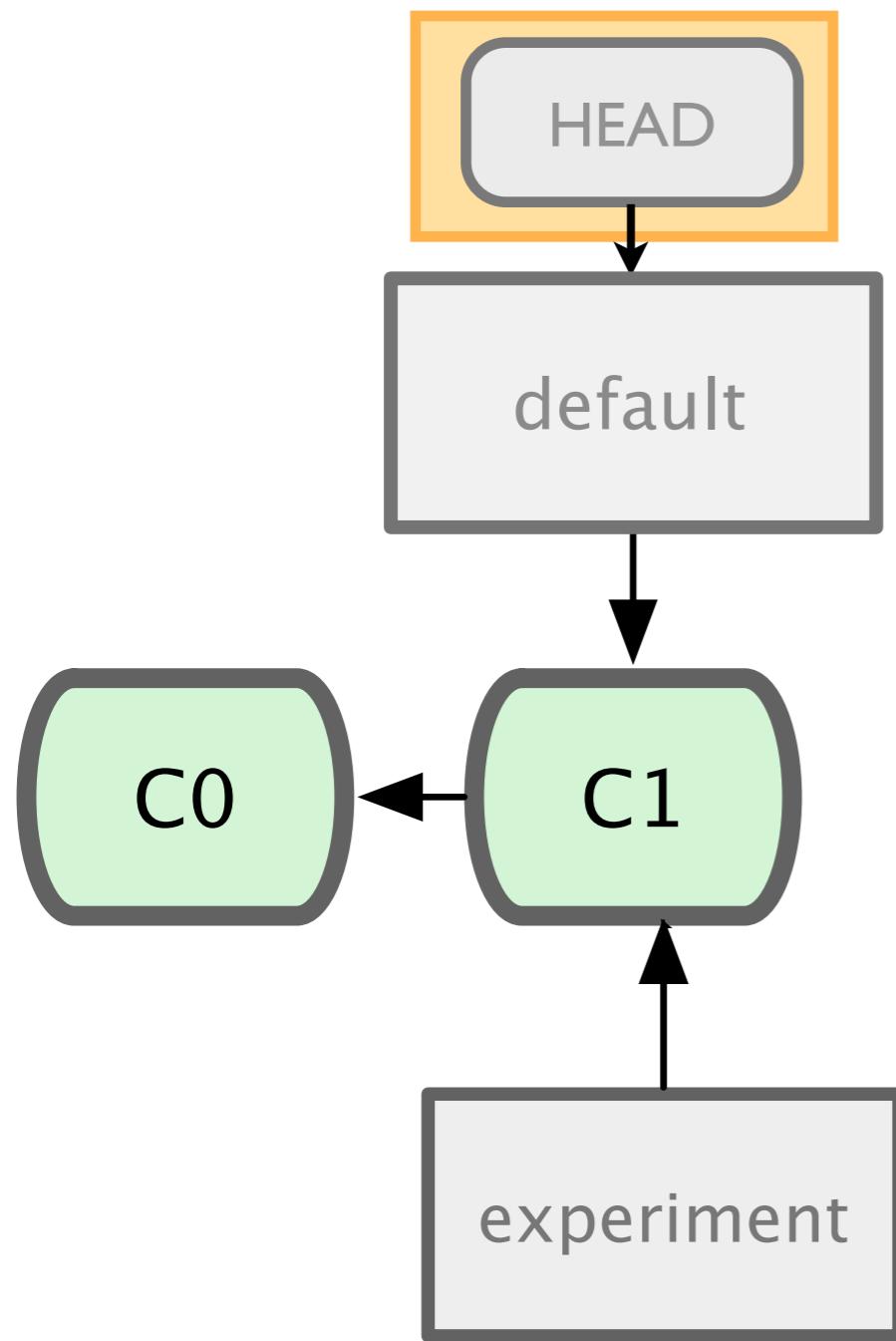
git branch experiment



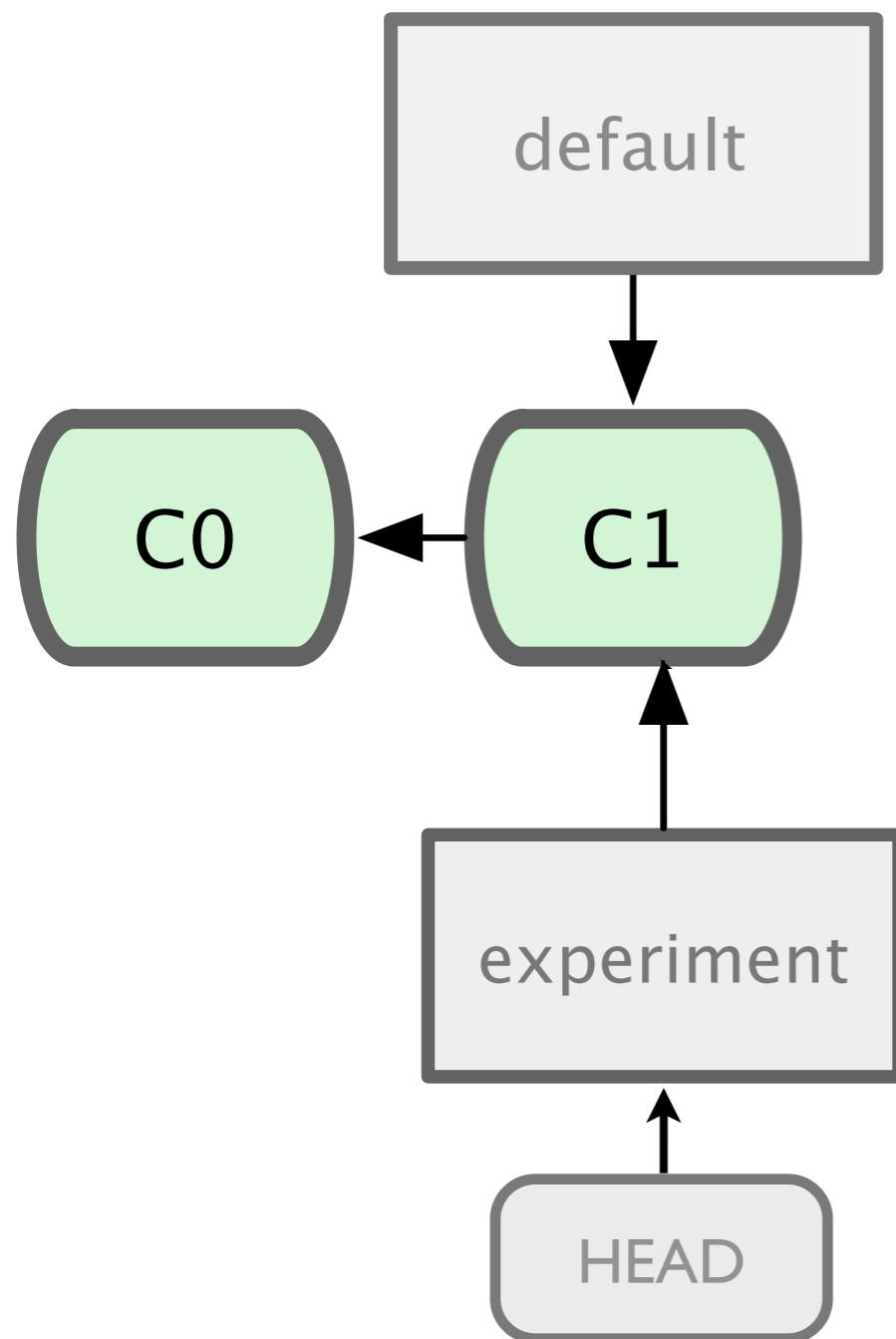
git branch experiment



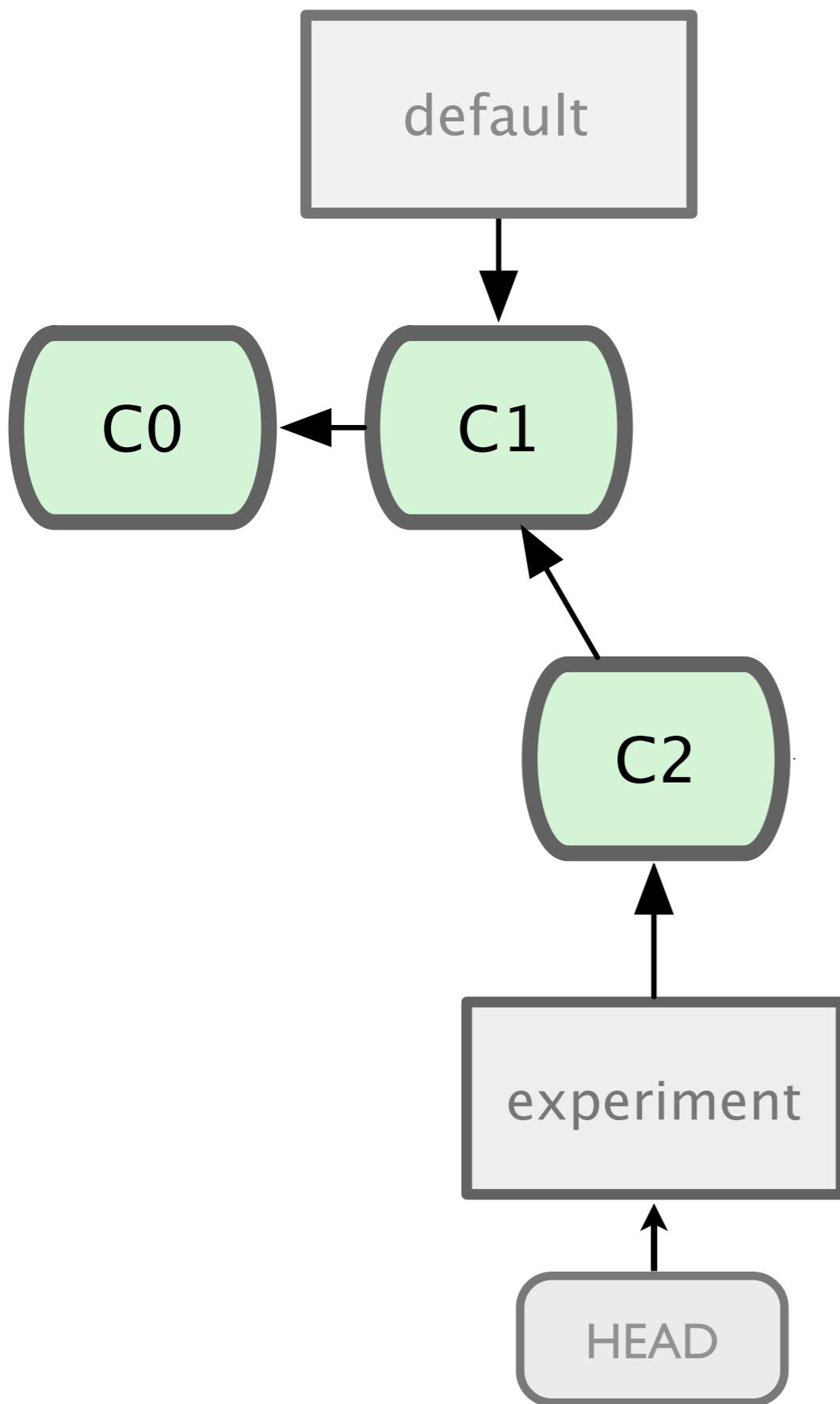
```
$ git branch  
* default  
  experiment
```



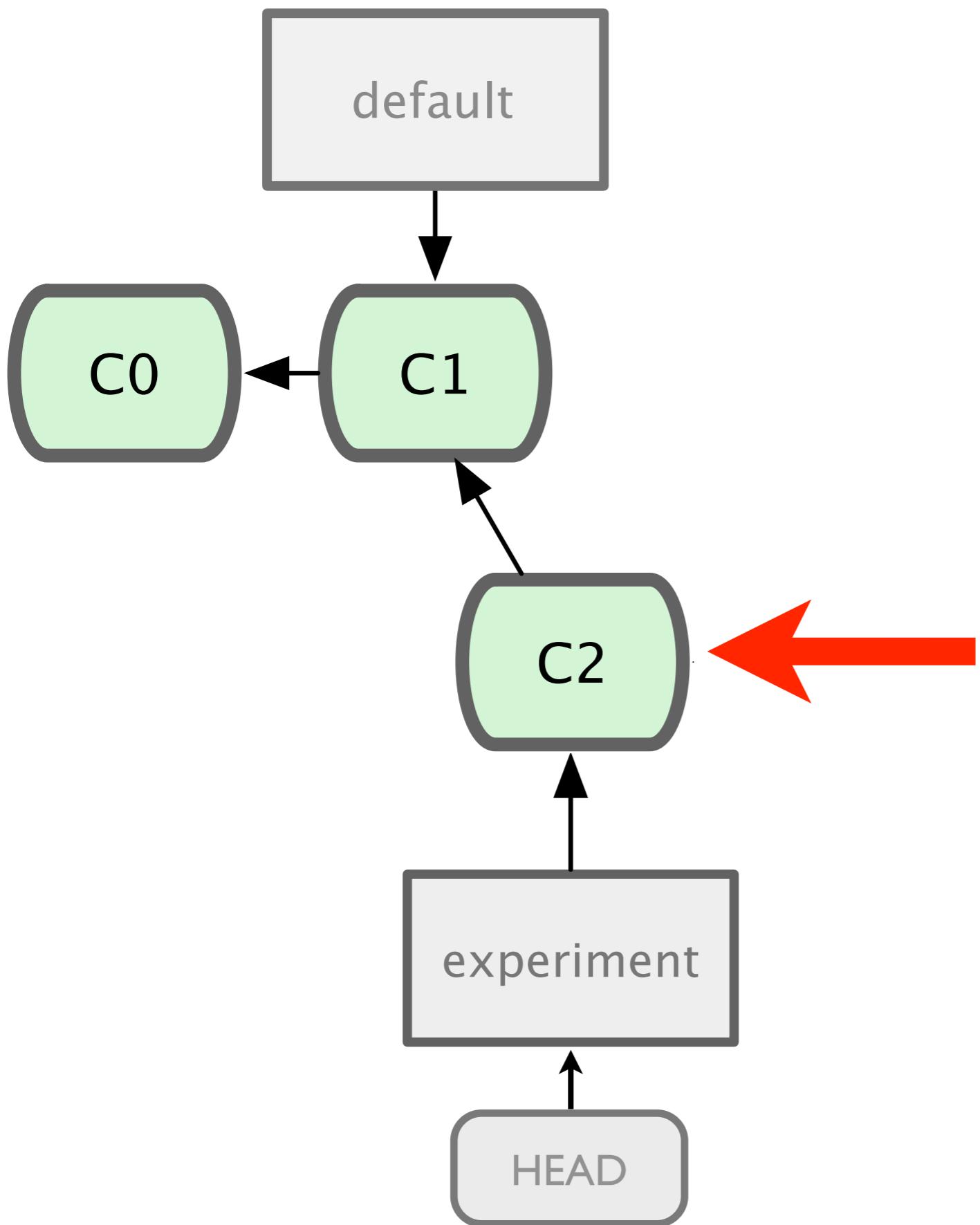
```
$ git branch  
* default  
  experiment
```



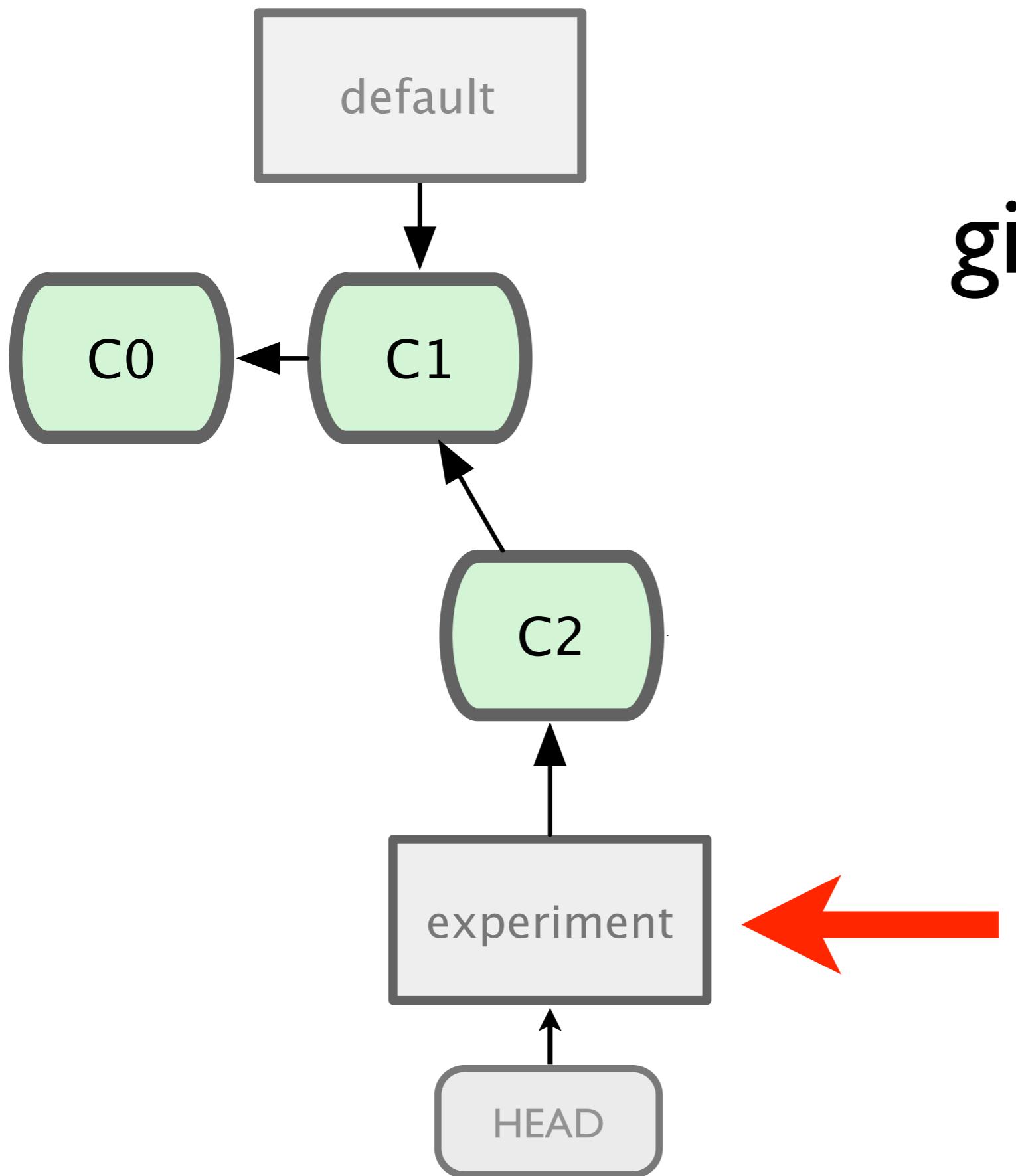
git checkout experiment



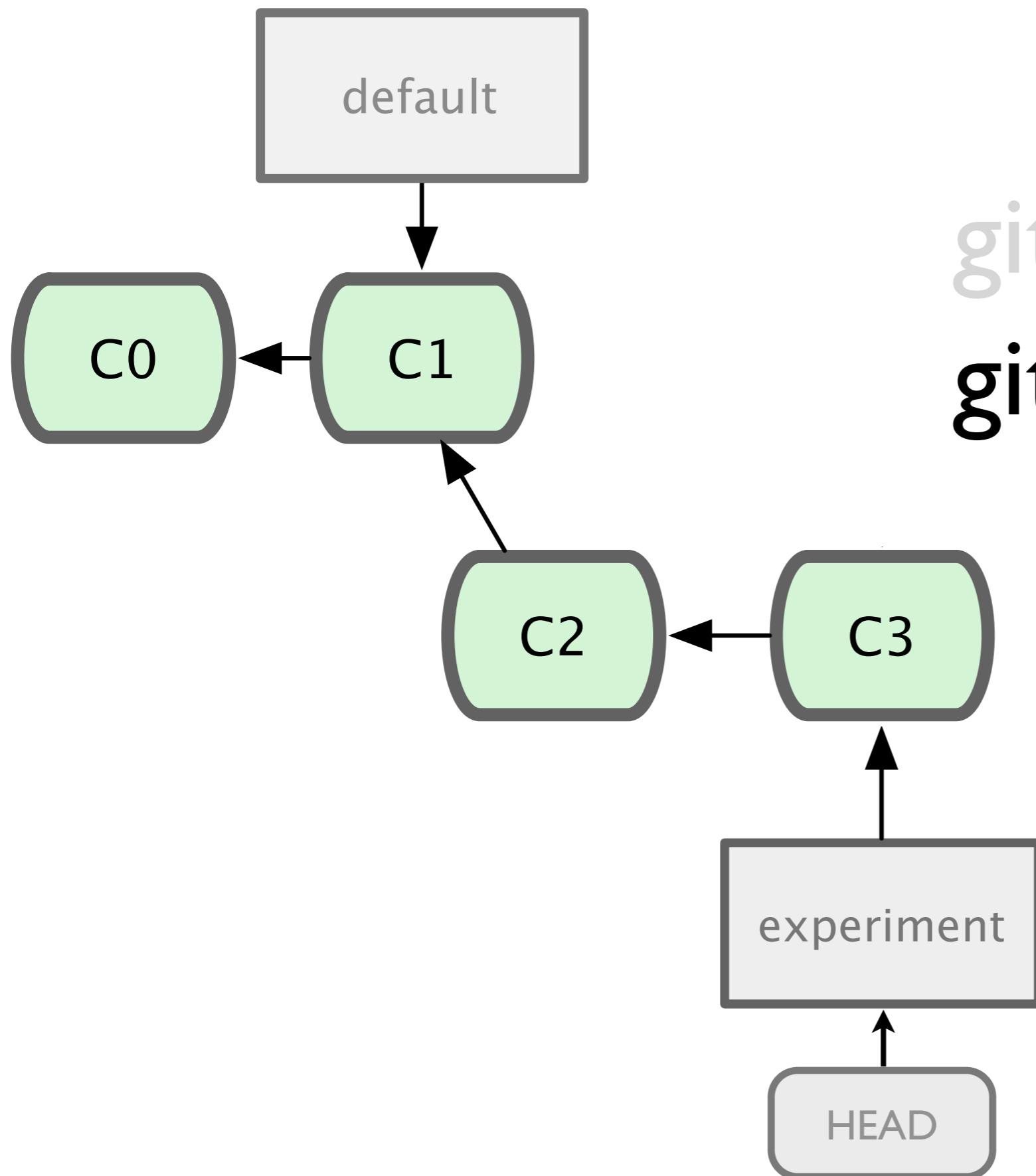
git commit



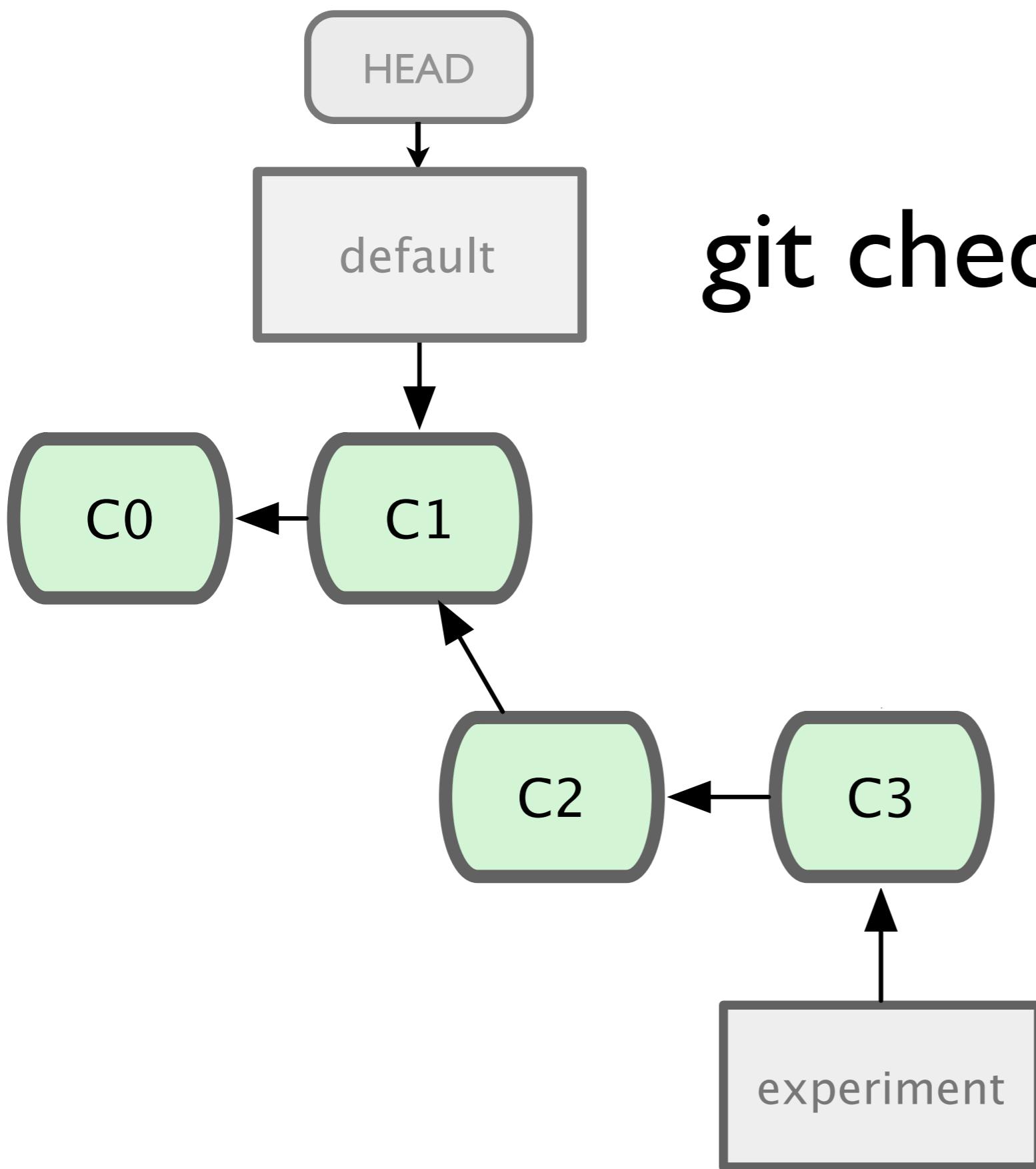
git commit



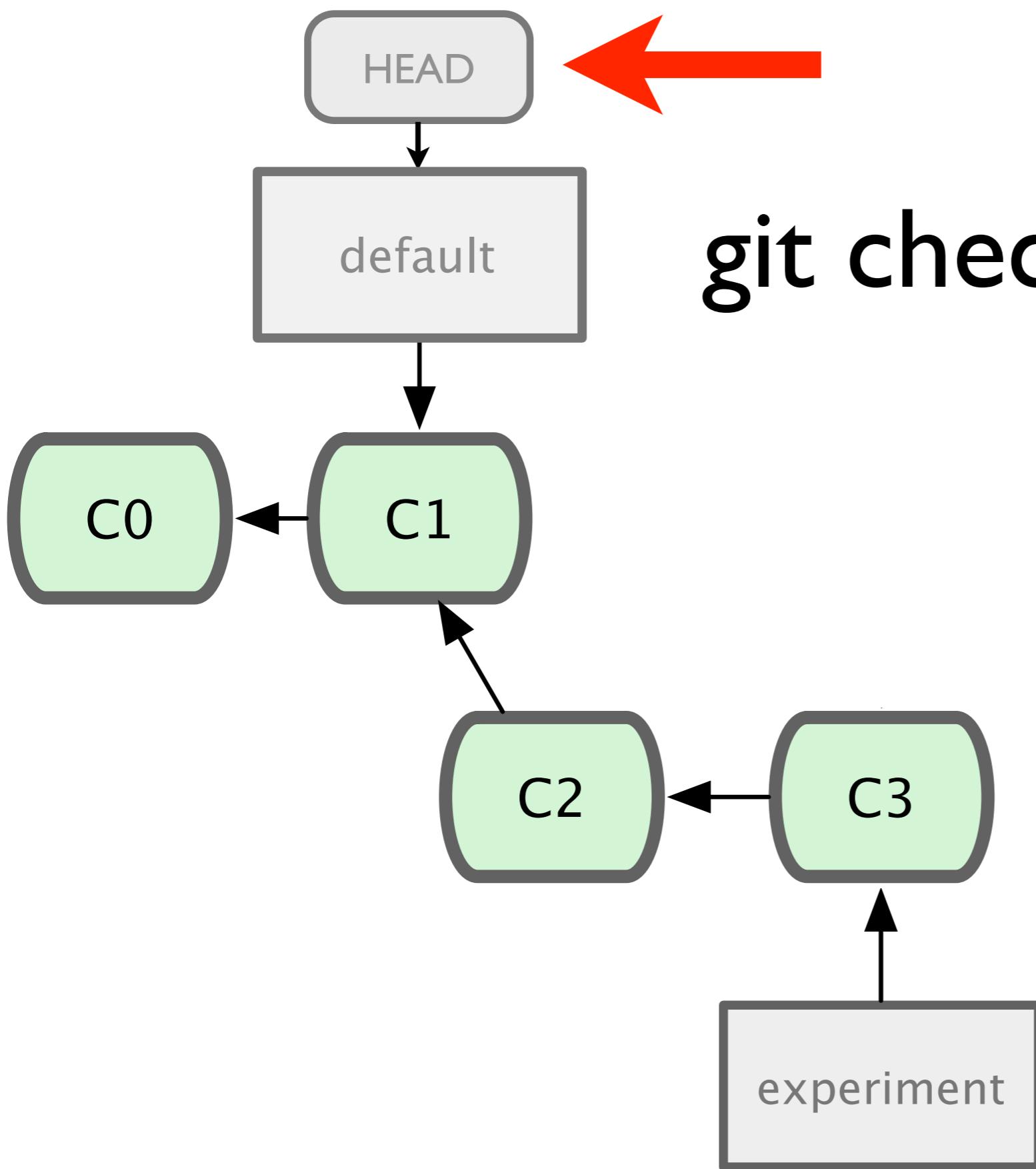
git commit



git commit
git commit

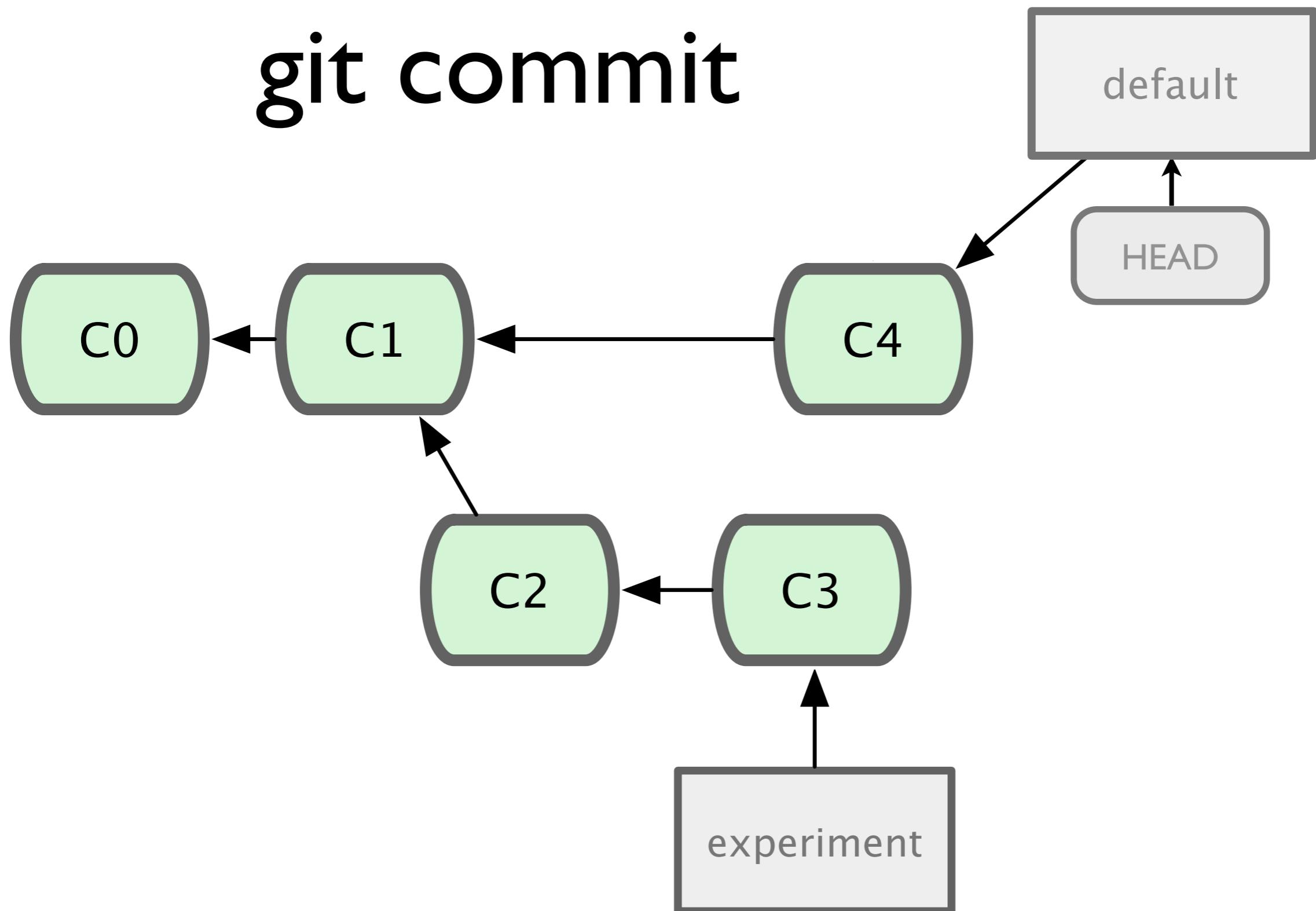


git checkout default

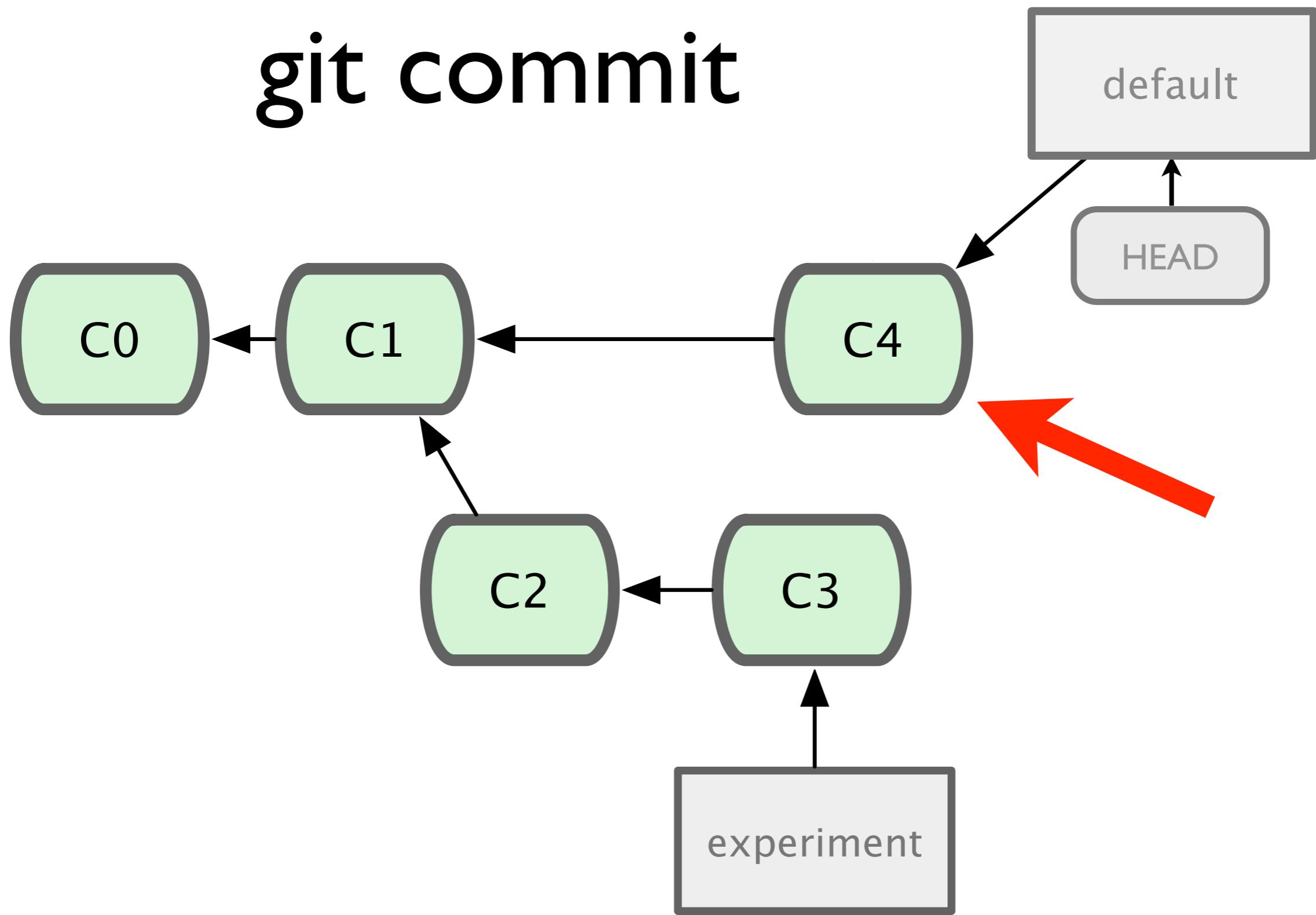


git checkout default

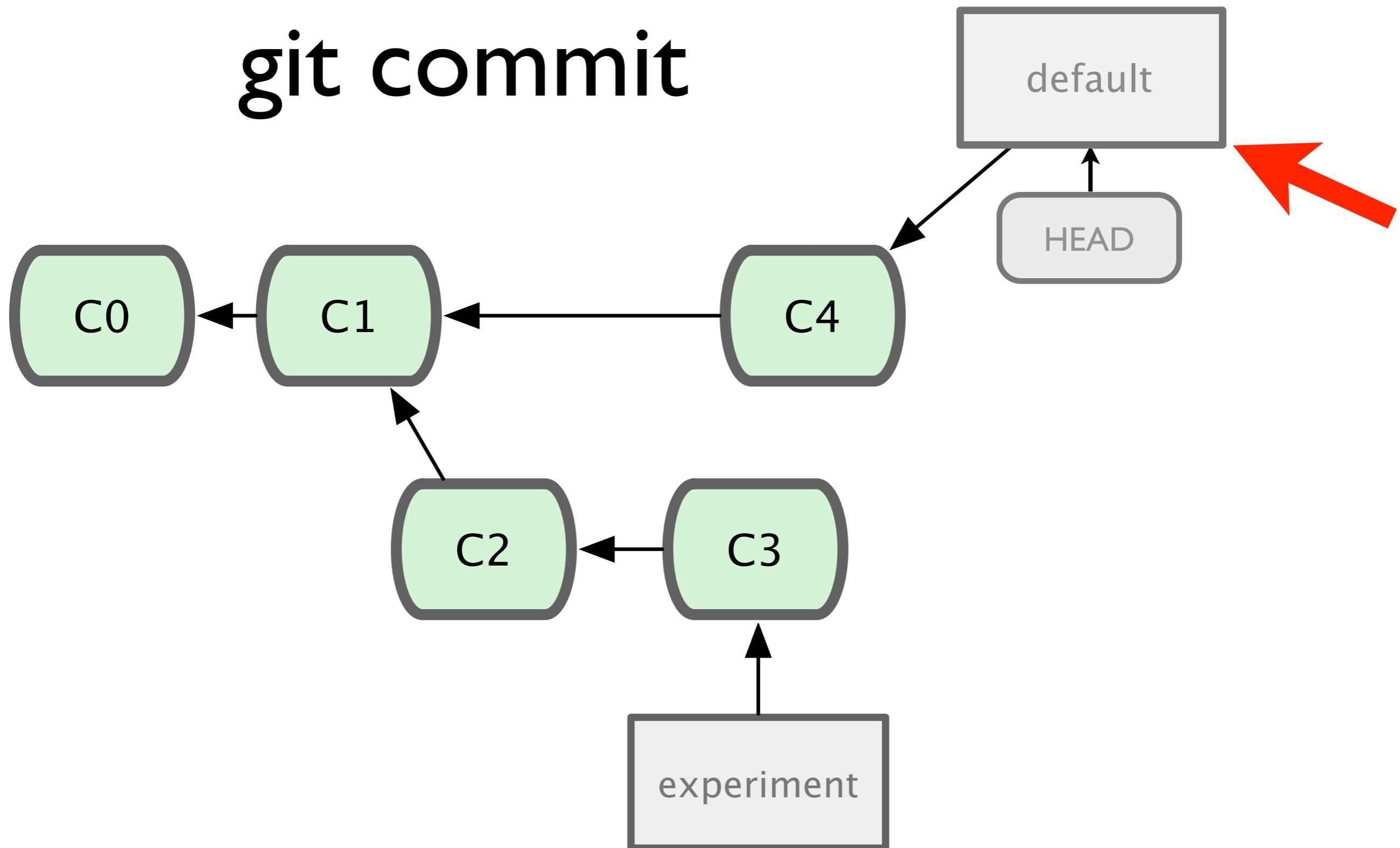
git commit

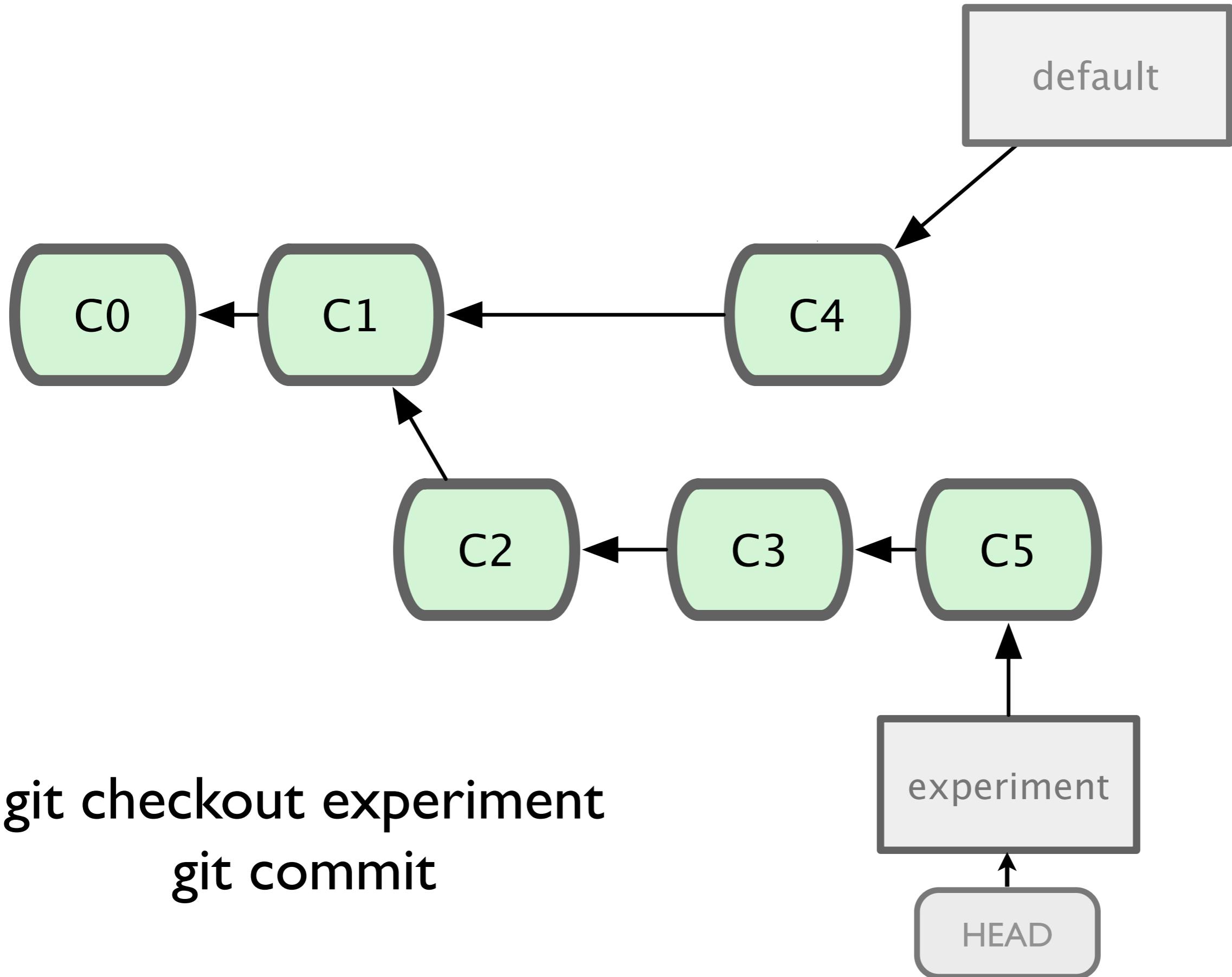


git commit



git commit





what is a branch, really?

```
$ find .git/refs
```

```
$ find .git/refs  
.git/refs  
.git/refs/heads  
.git/refs/heads/default  
.git/refs/heads/experiment
```

```
$ find .git/refs  
.git/refs  
.git/refs/heads  
.git/refs/heads/default  
.git/refs/heads/experiment
```

```
$ find .git/refs  
.git/refs  
.git/refs/heads  
/.git/refs/heads/default  
.git/refs/heads/experiment
```

```
$ cat .git/refs/heads/default  
6370409dc9e38af91565082bdf93577ff555489e
```

```
$ find .git/refs  
.git/refs  
.git/refs/heads  
.git/refs/heads/default  
.git/refs/heads/experiment
```

```
$ cat .git/refs/heads/default  
6370409dc9e38af91565082bd93577ff555489e  
$ cat .git/refs/heads/experiment  
33ce13c2f0ed35775956f81191b01a644448dcfc
```

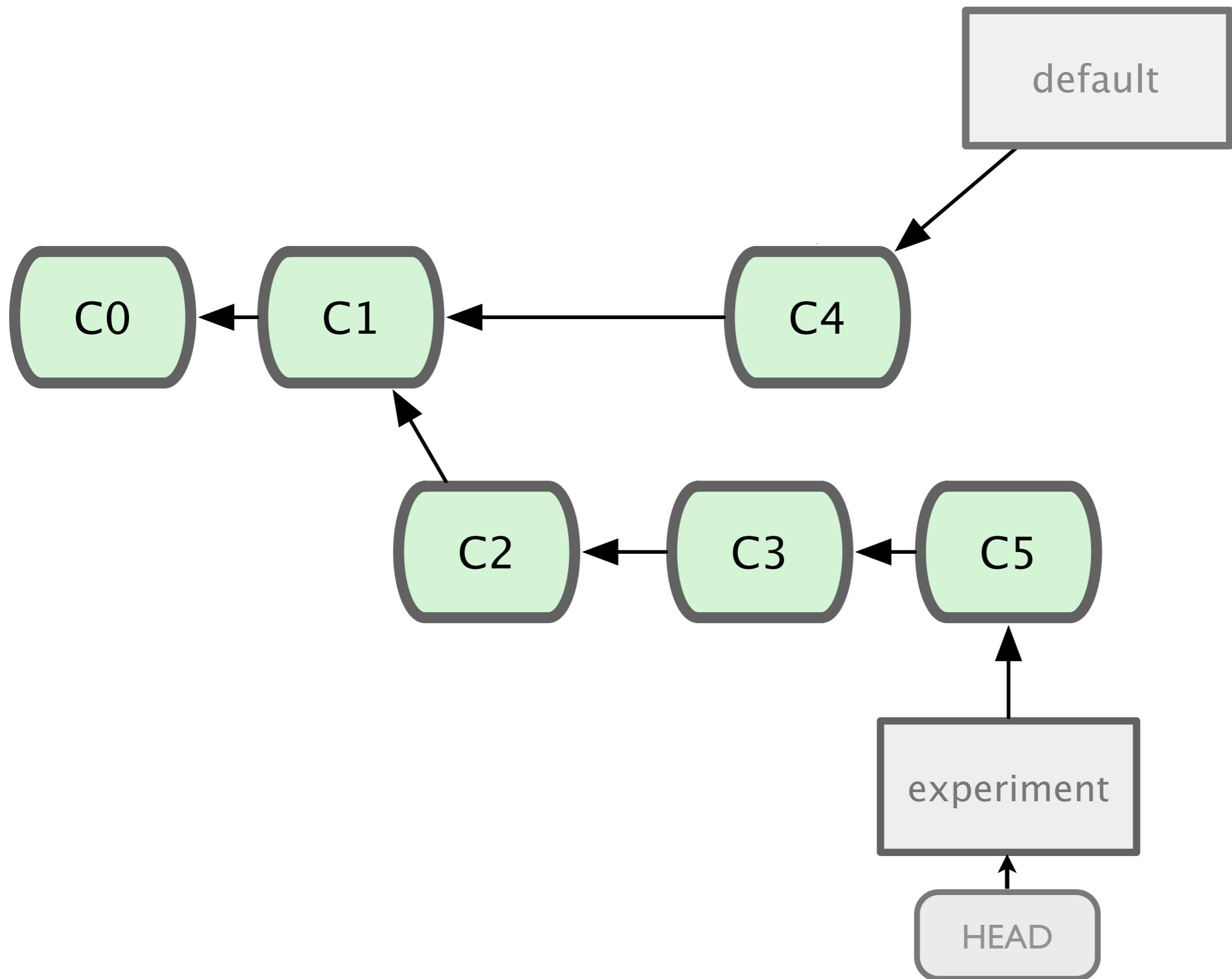
```
$ cat .git/HEAD
```

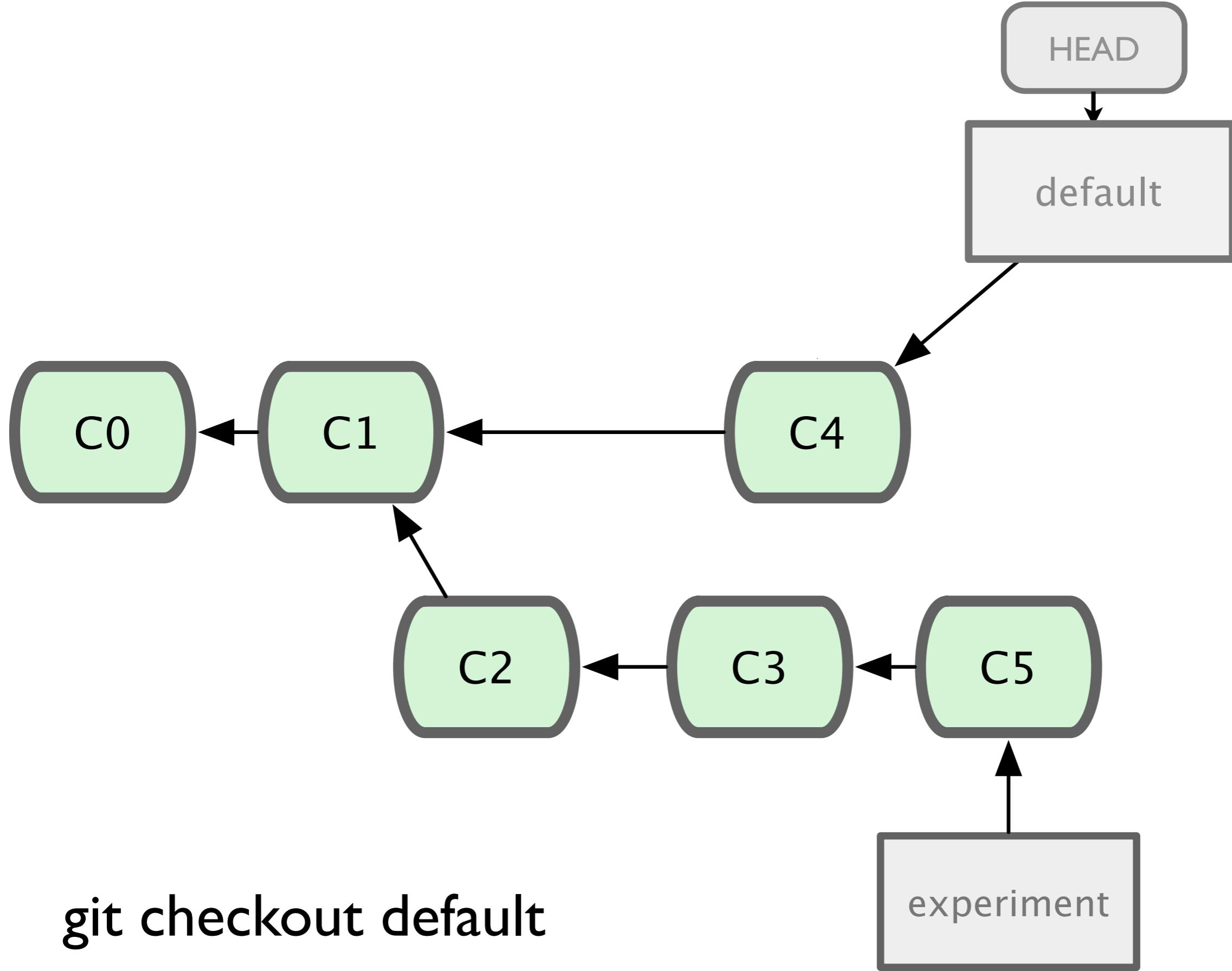
```
$ cat .git/HEAD  
ref: refs/heads/experiment
```

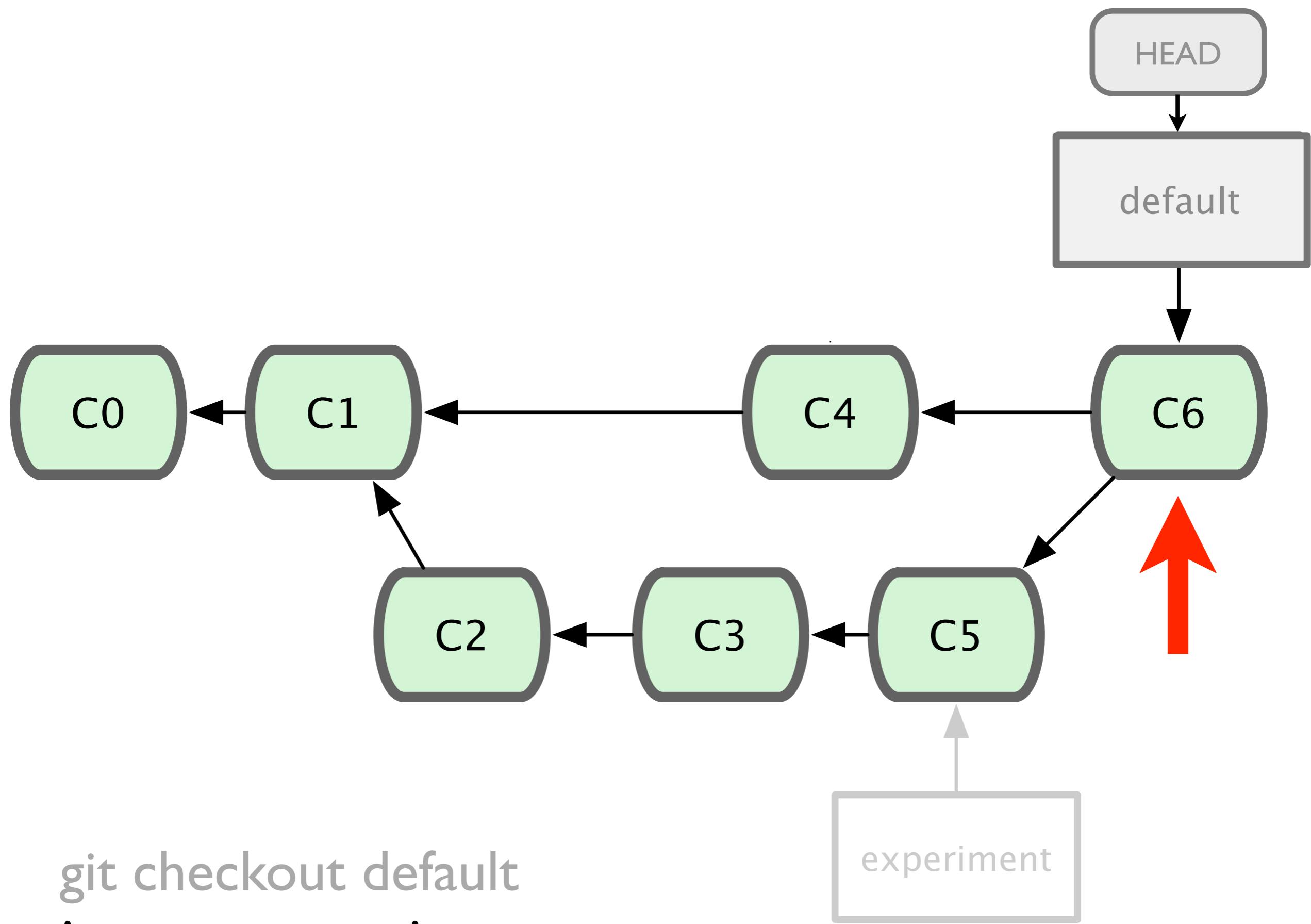
```
$ cat .git/HEAD  
ref: refs/heads/experiment
```

merging

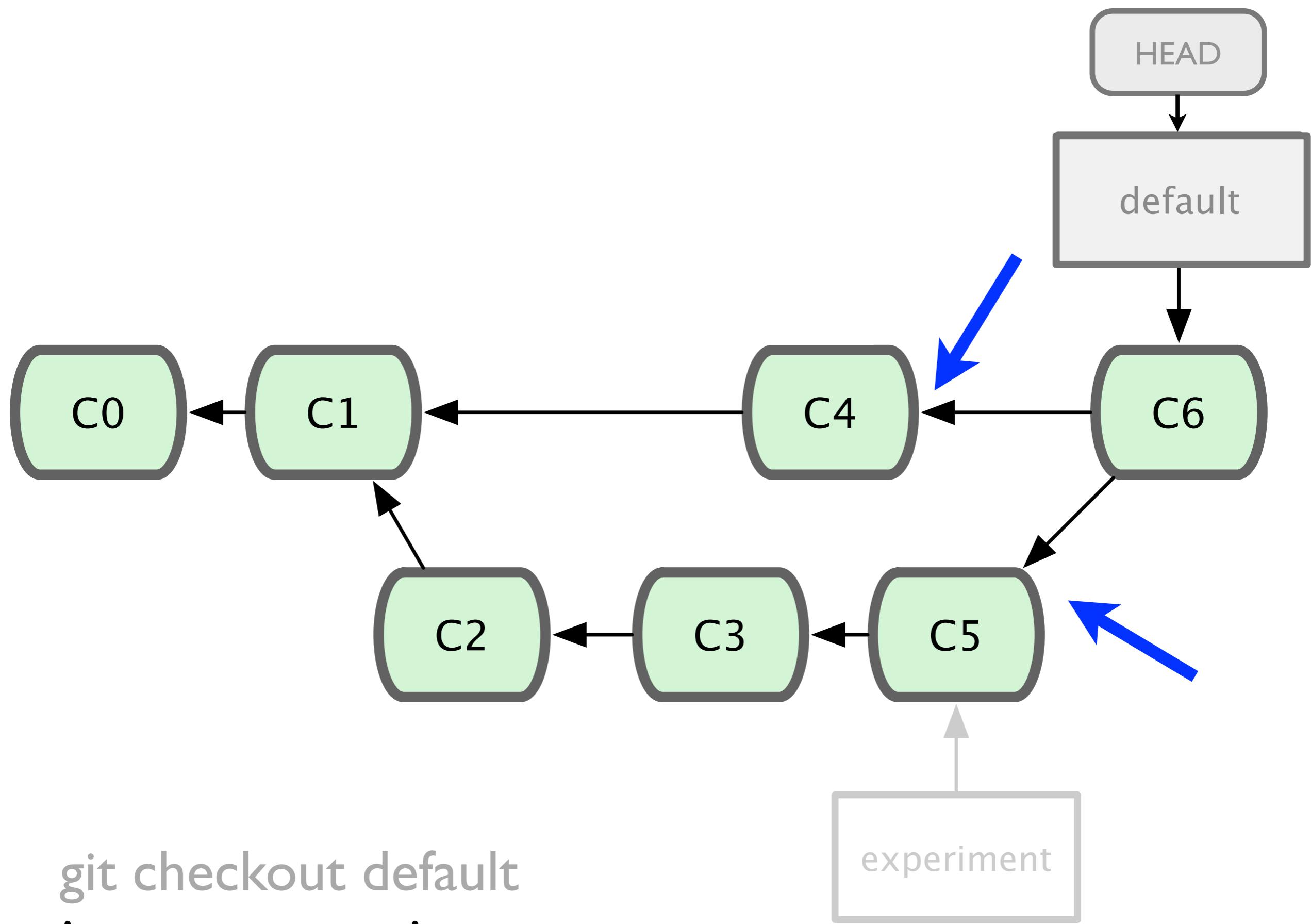
git merge

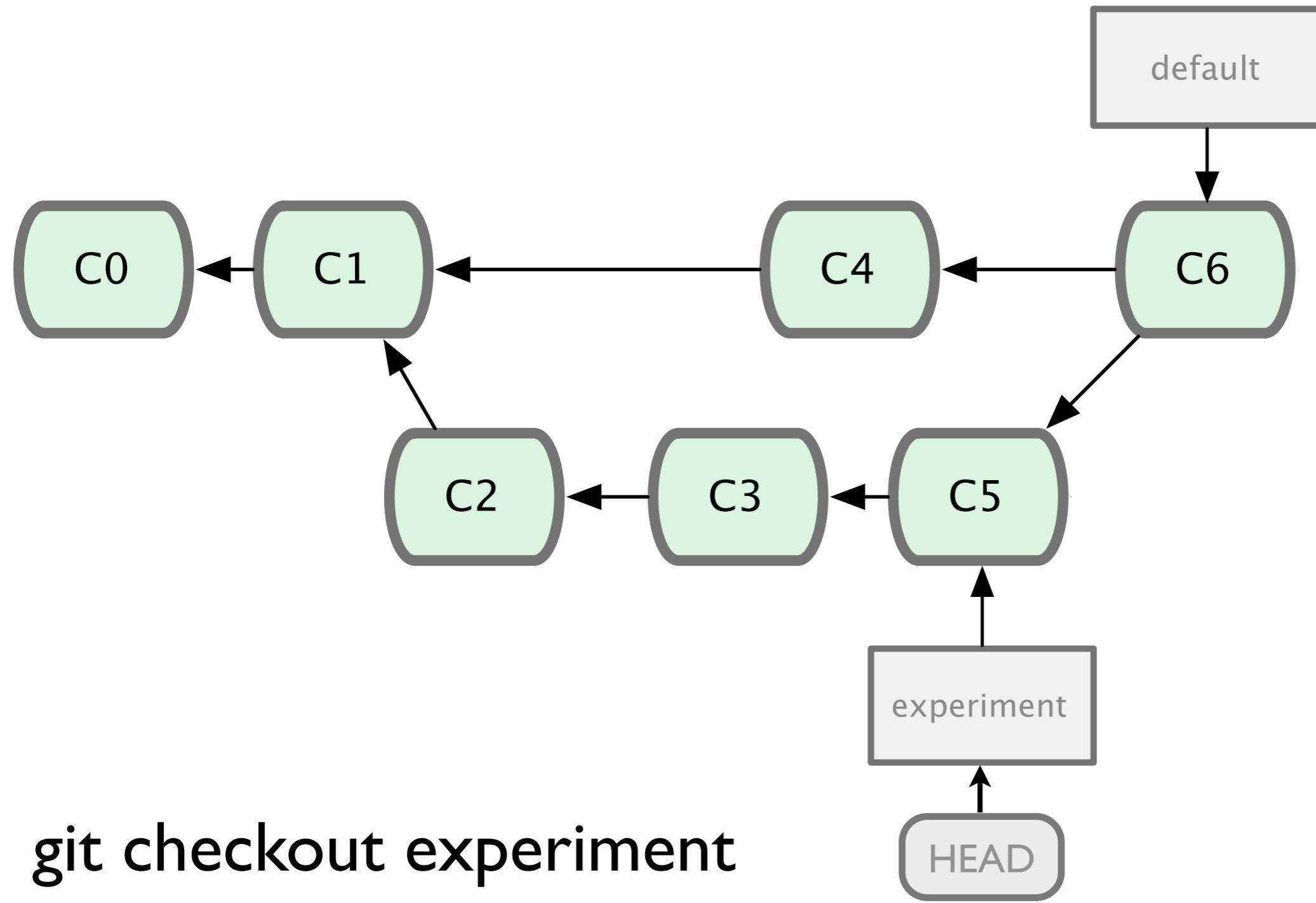


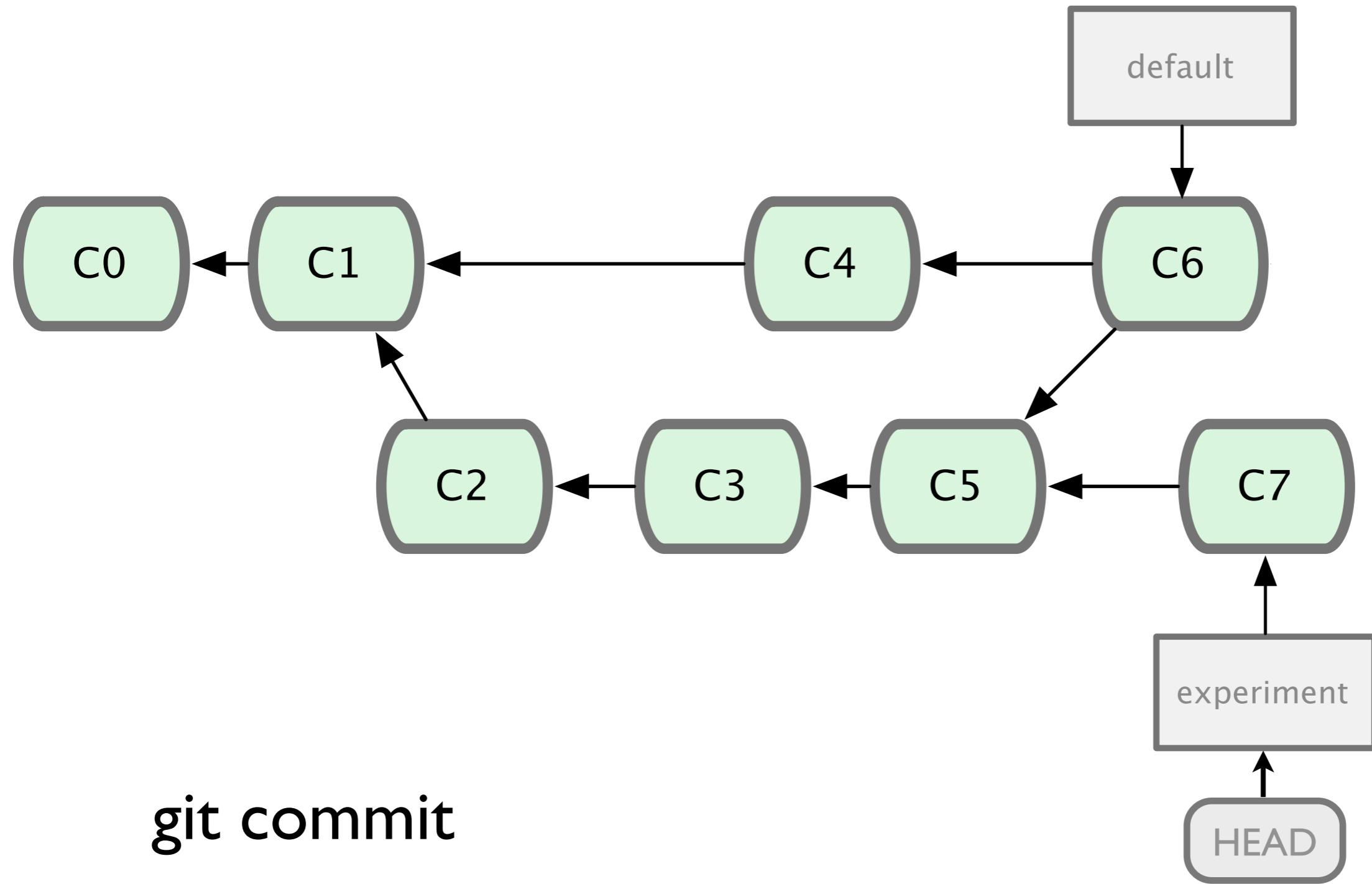


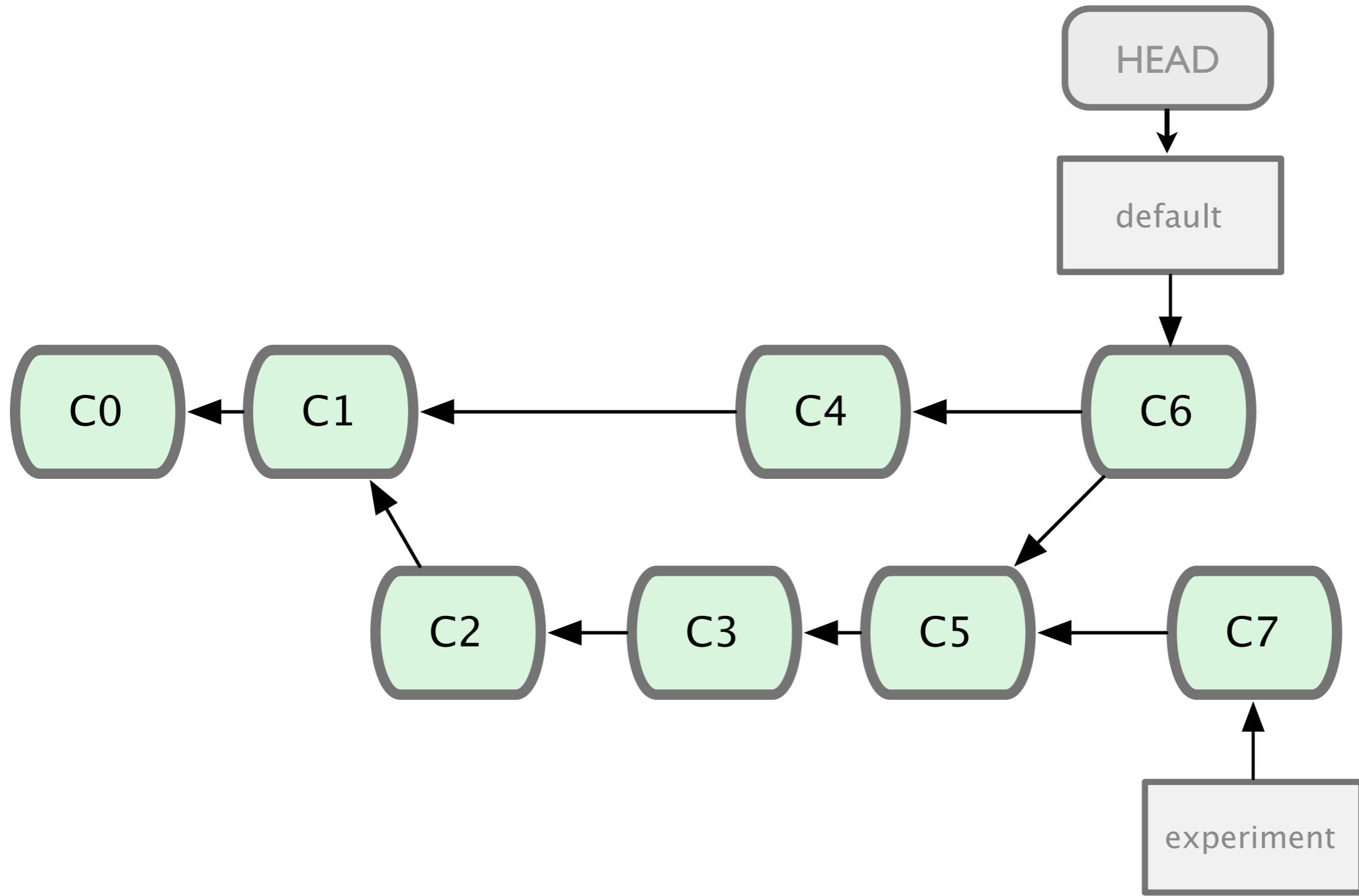


`git checkout default`
`git merge experiment`

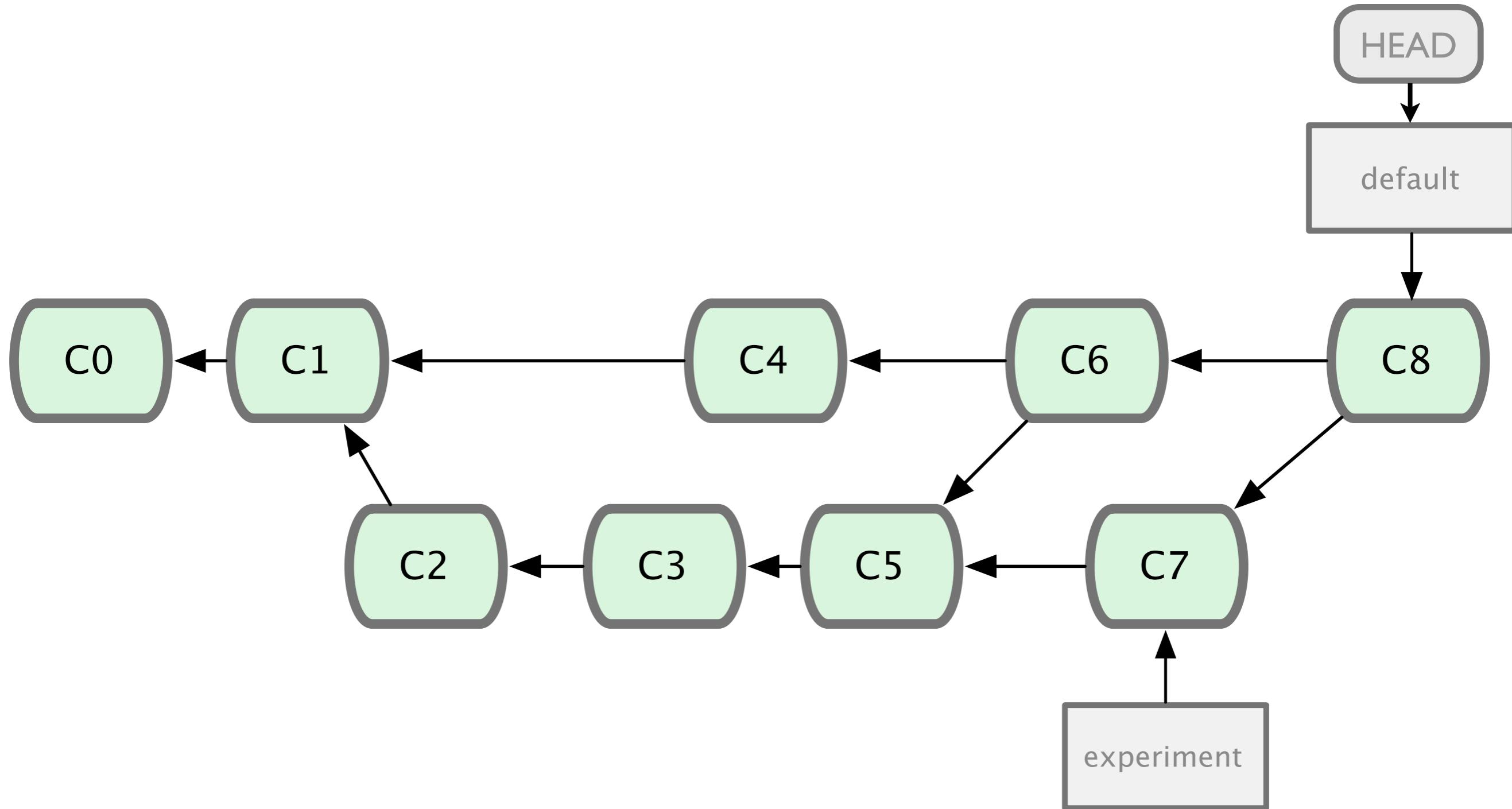








git checkout default



git merge experiment

Why is this cool?

try out an idea

isolate work units

long running topics

visualizing your
branches

`git log --graph`

git log --oneline --graph

```
* 20d4111 - fix copy strategy for production
* 86ef476 - confirm daemon servering is on for public imports
* 990bc8b - Merge branch 'friendly_delete'
  \
  * ed555b7 - better deletion flow
  * eec7491 - verify private forks aren't deleted
  * d65fc92 - simple fork deletion logic
  * e9370f4 - one more test for the gipper
  * 1d7c276 - remove busted test, add deleted template test
  * c0bf99f - let forks know their private network was deleted
* | ef2a5e2 - out of sight, out of mind
* | d352768 - guess we can turn this off now
* | d0f25c0 - use default timeout
* | 2e6eeae - convert executes to AR:B
* | e6b8628 - Merge branch 'mirrors'
  \
  * | ca3744f - couple mirror bug fixes
  * | 56b8868 - mirroring
* | | 08cb112 - bang. ya dead.
  /
  * | c153b21 - let AR:B figure it out
  * | 19e66a2 - raw private gists are ssl'd
  * | fdda4ac - Merge branch 'master' of git@github.com:defunkt/github
  \
  * | 0e9cf1 - x add slice39 to the deploy role
  * | 82703eb - Revert "update newrelic plugin"
  * | 77faff1 - turn on tracer
  * | 3807239 - update newrelic plugin
  * | 0aaehh2 - sexy up inbox messages
```

gitk

gitk: github

fix copy strategy for production
confirm daemon servering is on for public imports
Merge branch 'friendly_delete'
better deletion flow
verify private forks aren't deleted
simple fork deletion logic
one more test for the gipper
remove busted test, add deleted template test
let forks know their private network was deleted
out of sight, out of mind
guess we can turn this off now
use default timeout
convert executes to AR:B
Merge branch 'mirrors'
couple mirror bug fixes
mirroring
bang. ya dead.
let AR:B figure it out
raw private gists are ssl'd
Merge branch 'master' of git@github.com:defunkt/github
x add slice39 to the deploy role
Report "undate no url fix plugin"

PJ Hyett <pjhyett@gmail.com> 2008-09-30 16:49:34
PJ Hyett <pjhyett@gmail.com> 2008-09-29 18:17:19
PJ Hyett <pjhyett@gmail.com> 2008-09-29 16:07:04
PJ Hyett <pjhyett@gmail.com> 2008-09-29 16:05:17
PJ Hyett <pjhyett@gmail.com> 2008-09-29 15:31:02
PJ Hyett <pjhyett@gmail.com> 2008-09-29 15:13:51
PJ Hyett <pjhyett@gmail.com> 2008-09-22 17:03:25
PJ Hyett <pjhyett@gmail.com> 2008-09-22 16:52:40
PJ Hyett <pjhyett@gmail.com> 2008-09-19 18:31:50
PJ Hyett <pjhyett@gmail.com> 2008-09-29 12:46:18
PJ Hyett <pjhyett@gmail.com> 2008-09-29 12:09:16
PJ Hyett <pjhyett@gmail.com> 2008-09-29 11:57:49
PJ Hyett <pjhyett@gmail.com> 2008-09-26 17:46:50
PJ Hyett <pjhyett@gmail.com> 2008-09-26 15:32:34
PJ Hyett <pjhyett@gmail.com> 2008-09-26 15:32:09
PJ Hyett <pjhyett@gmail.com> 2008-09-26 15:17:37
Chris Wanstrath <chris@ozmm.org> 2008-09-26 14:46:15
PJ Hyett <pjhyett@gmail.com> 2008-09-24 18:29:54
Chris Wanstrath <chris@ozmm.org> 2008-09-24 14:31:31
Chris Wanstrath <chris@ozmm.org> 2008-09-23 18:48:04
PJ Hyett <pjhyett@gmail.com> 2008-09-23 17:35:29
PJ Hyett <pjhyett@gmail.com> 2008-09-23 17:28:58

SHA1 ID: 20d4113a586b0ed05f530e6f7f4b895540da0f0

Find next prev commit containing: Exact All fields

Search Patch Tree

Diff Old version New version Lines of context: 3 Ignore space change

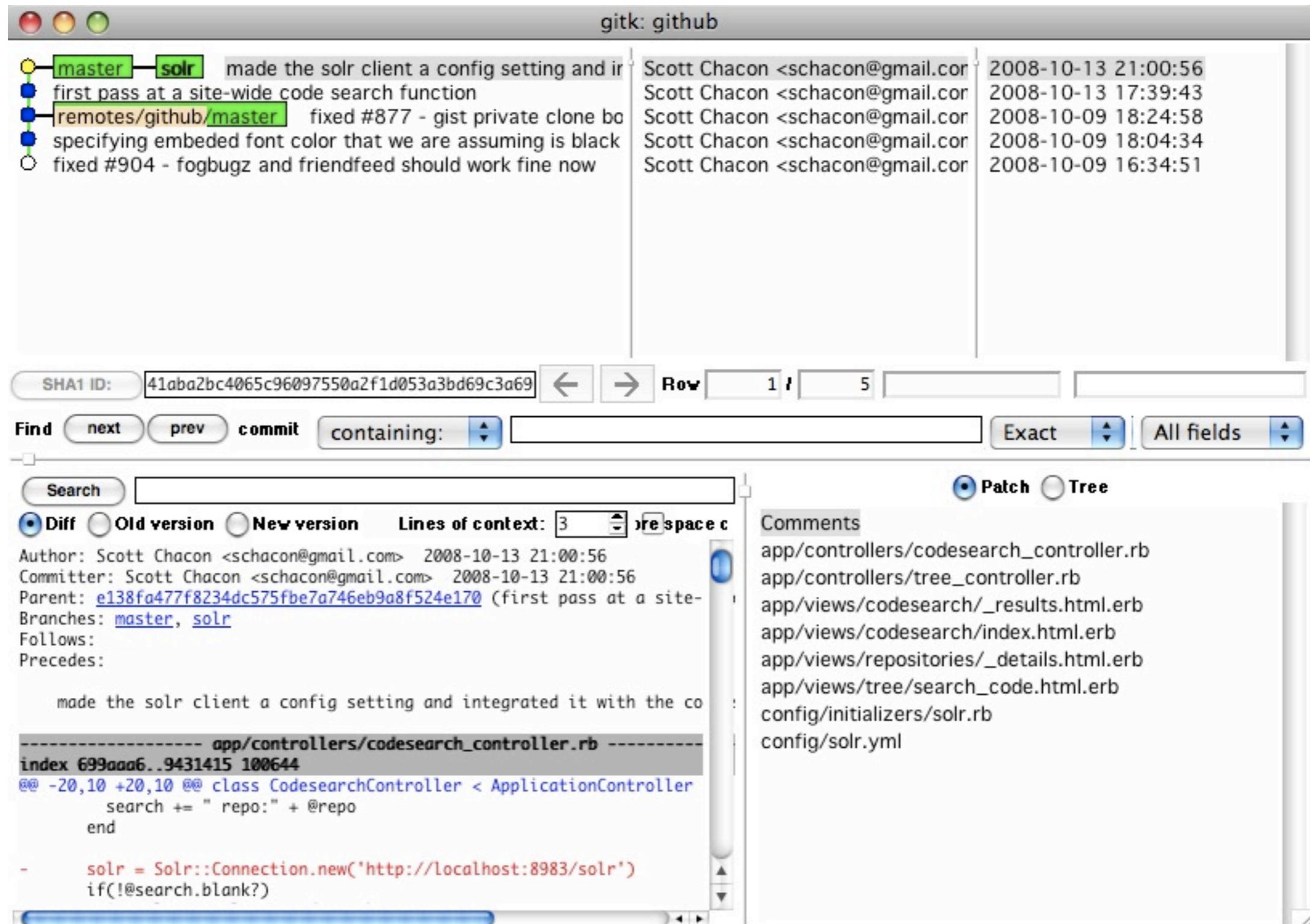
Author: PJ Hyett <pjhyett@gmail.com> 2008-09-30 16:49:34
Committer: PJ Hyett <pjhyett@gmail.com> 2008-09-30 16:49:34
Parent: [86ef47642bf2a1c67820de999e15437601e872bb](#) (confirm daemon servering is on for public
Branches: master, remotes/github/master, solr, training, wiki2
Follows:
Precedes:

fix copy strategy for production

----- app/models/repository.rb -----
index 09b8659..574ab6b 100644
@@ -82,11 +82,7 @@ class Repository < ActiveRecord::Base
end

Comments
app/models/repository.rb
jobs/repository-delete
test/unit/repository_delete_job_test.rb

gitk 6d13f..41aba



git log --all

Deleting Branches

```
git branch -d branch
```

```
git branch -D branch
```

Stashing

**when a commit is
too much**

```
$ git status
```

```
$ git status
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git status
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git stash
```

```
$ git status
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git stash
Saved working directory and index state "WIP on iss53: c8208f9...
Merge branch 'cupcake'"
HEAD is now at c8208f9 Merge branch 'cupcake'
(To restore them type "git stash apply")
```

```
$ git status
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git stash
Saved working directory and index state "WIP on iss53: c8208f9...
Merge branch 'cupcake'"
HEAD is now at c8208f9 Merge branch 'cupcake'
(To restore them type "git stash apply")
```

```
$ git status
# On branch iss53
nothing to commit (working directory clean)
```

```
$ git status
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git stash
Saved working directory and index state "WIP on iss53: c8208f9...
Merge branch 'cupcake'"
HEAD is now at c8208f9 Merge branch 'cupcake'
(To restore them type "git stash apply")
```

```
$ git status
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git stash
Saved working directory and index state "WIP on iss53: c8208f9...
Merge branch 'cupcake'"
HEAD is now at c8208f9 Merge branch 'cupcake'
(To restore them type "git stash apply")
```

```
$ git status
```

```
$ git status
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git stash
Saved working directory and index state "WIP on iss53: c8208f9...
Merge branch 'cupcake'"
HEAD is now at c8208f9 Merge branch 'cupcake'
(To restore them type "git stash apply")
```

```
$ git status
# On branch iss53
nothing to commit (working directory clean)
```

```
$ git status
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git stash
Saved working directory and index state "WIP on iss53: c8208f9...
Merge branch 'cupcake'"
HEAD is now at c8208f9 Merge branch 'cupcake'
(To restore them type "git stash apply")
```

```
$ git status
# On branch iss53
nothing to commit (working directory clean)
```

```
$ git stash list
```

```
$ git stash list  
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'
```

```
$ git stash list  
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'
```

```
$ git stash apply
```

```
$ git stash list
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'

$ git stash apply
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
#       modified:   src/com/android/alarmclock/AlarmAlert.java
#       modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

```
$ git stash list
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'

$ git stash apply
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")

$ git stash list
```

```
$ git stash list
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'

$ git stash apply
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")

$ git stash list
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'
```

```
$ git stash list
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'

$ git stash apply
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")

$ git stash list
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'

$ git stash drop stash@{0}
```

```
$ git stash list
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'

$ git stash apply
# On branch iss53
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working
#    directory)
#
# modified:   src/com/android/alarmclock/AlarmAlert.java
# modified:   src/com/android/alarmclock/AlarmAlertWakeLock.java
#
no changes added to commit (use "git add" and/or "git commit -a")

$ git stash list
stash@{0}: WIP on iss53: c8208f9... Merge branch 'cupcake'

$ git stash drop stash@{0}
Dropped stash@{0} (e19b0e94032d0614f450238f91953fe5e7e454a8)
```

```
git stash branch new_branch stash@{1}
```

git stash

save

list

drop

pop

apply

branch

Remotes

sharing git

```
$>git remote  
$>
```



schacon



0

[account](#) | [profile](#) | [guides](#) | [log out](#)
[repositories: all](#) | [search](#)

News Feed

for you | from you

 **Magnus Chacon committed to simplegit 4 minutes ago**



[fc62e5543b195f18391886b9f663d5a7eca38e84](#)

added file_size

 **Magnus Chacon committed to simplegit 27 minutes ago**



[f40831985e265eb8cca2d0e66c11ac48ccfe170d](#)

changes

 **Magnus Chacon committed to simplegit 27 minutes ago**



[25ce6394a94d8c3c30240960b386f79f4ebdd1da](#)

changes

 **Magnus Chacon committed to simplegit 27 minutes ago**



[3e625e2139a71bfbaa85b055c139bceafab7e2c0](#)

updated rakefile

 **Magnus Chacon committed to simplegit 27 minutes ago**



[4fefbaafdf51ff3e79b84e9ee1a0dac2848e21a98](#)

commit the todo

Your Repositories [\(create a new one\)](#)

[all](#) | [public](#) | [private](#) | [sources](#) | [forks](#)

 [capistrano](#)

 [dst](#)

 [facebook](#)

 [git-ruby](#)

 [git-source](#)

 [git-wiki](#)

 [grit](#)

 [munger](#)

 [rest-client](#)

 [ruby-git](#)

 [simplegit](#)

 [ticgit](#)

Watched Repositories



schacon



0

[account](#) | [profile](#) | [guides](#) | [log out](#)

[repositories: all](#) | [search](#)

News Feed

[for you](#) | [from you](#)

 **Magnus Chacon committed to simplegit 4 minutes ago**



[fc62e5543b195f18391886b9f663d5a7eca38e84](#)

added file_size

 **Magnus Chacon committed to simplegit 27 minutes ago**



[f40831985e265eb8cca2d0e66c11ac48ccfe170d](#)

changes

 **Magnus Chacon committed to simplegit 27 minutes ago**



[25ce6394a94d8c3c30240960b386f79f4ebdd1da](#)

changes

 **Magnus Chacon committed to simplegit 27 minutes ago**



[3e625e2139a71bfbaa85b055c139bceafab7e2c0](#)

updated rakefile

 **Magnus Chacon committed to simplegit 27 minutes ago**



[4fefbaafdf51ff3e79b84e9ee1a0dac2848e21a98](#)

commit the todo

Your Repositories (create a new one)

[all](#) | [public](#) | [private](#) | [sources](#) | [forks](#)

 [capistrano](#)

 [dst](#)

 [facebook](#)

 [git-ruby](#)

 [git-source](#)

 [git-wiki](#)

 [grit](#)

 [munger](#)

 [rest-client](#)

 [ruby-git](#)

 [simplegit](#)

 [ticgit](#)

Watched Repositories

Create a New Repository

Create a new empty repository into which you can push your local git repo.

NOTE: If you intend to push a copy of a repository that is already hosted on GitHub, then you should [fork](#) it instead.

Project Name

Description

Homepage URL

Who has access to this repository? (You can change this later)

- Anyone ([learn more about public repos](#))
- Only the people I specify ([learn more about private repos](#))

[Submit](#)

**schacon**

✉ 0

[account](#) | [profile](#) | [guides](#) | [log out](#)[repositories: all](#) | [search](#)**schacon / simplegit2** [edit](#) **this repo is viewable by everyone**Description: another repo for simplegit for demo purposes [edit](#)Homepage: [Click to edit](#) [edit](#)Public Clone URL: <git://github.com/schacon/simplegit2.git>Your Clone URL: <git@github.com:schacon/simplegit2.git>Push URL: <git@github.com:schacon/simplegit2.git>

**schacon**

✉ 0

[account](#) | [profile](#) | [guides](#) | [log out](#)[repositories: all](#) | [search](#)**schacon / simplegit2** [edit](#) **this repo is viewable by everyone**Description: another repo for simplegit for demo purposes [edit](#)Homepage: [Click to edit](#) [edit](#)Public Clone URL: <git://github.com/schacon/simplegit2.git>Your Clone URL: <git@github.com:schacon/simplegit2.git>Push URL: <git@github.com:schacon/simplegit2.git>

```
$>git remote  
$>git remote add public git@github.com:schacon/simplegit2.git  
$>
```

```
$>git remote  
$>git remote add public git@github.com:schacon/simplegit2.git  
$>
```

```
$>git remote  
$>git remote add public git@github.com:schacon/simplegit2.git  
$>
```

```
$>git remote
$>git remote add public git@github.com:schacon/simplegit2.git
$>git push public master
Counting objects: 115, done.
Compressing objects: 100% (73/73), done.
Writing objects: 100% (115/115), 11.20 KiB, done.
Total 115 (delta 42), reused 65 (delta 24)
refs/heads/master: 000000000000000000000000000000000000000000000000000000000000000 -> fc6
5543b195f18391886b9f663d5a7eca38e84
To git@github.com:schacon/simplegit2.git
 * [new branch]           master -> master
$>
```

```
$>git remote
$>git remote add public git@github.com:schacon/simplegit2.git
$>git push public master
Counting objects: 115, done.
Compressing objects: 100% (73/73), done.
Writing objects: 100% (115/115), 11.20 KiB, done.
Total 115 (delta 42), reused 65 (delta 24)
refs/heads/master: 000000000000000000000000000000000000000000000000000000000000000 -> fc6
5543b195f18391886b9f663d5a7eca38e84
To git@github.com:schacon/simplegit2.git
 * [new branch]           master -> master
$>
```

```
$>git remote
$>git remote add public git@github.com:schacon/simplegit2.git
$>git push public master
Counting objects: 115, done.
Compressing objects: 100% (73/73), done.
Writing objects: 100% (115/115), 11.20 KiB, done.
Total 115 (delta 42), reused 65 (delta 24)
refs/heads/master: 000000000000000000000000000000000000000000000000000000000000000 -> fc6
5543b195f18391886b9f663d5a7eca38e84
To git@github.com:schacon/simplegit2.git
 * [new branch]          master -> master
$>
```

```
$>git remote
$>git remote add public git@github.com:schacon/simplegit2.git
$>git push public master
Counting objects: 115, done.
Compressing objects: 100% (73/73), done.
Writing objects: 100% (115/115), 11.20 KiB, done.
Total 115 (delta 42), reused 65 (delta 24)
refs/heads/master: 000000000000000000000000000000000000000000000000000000000000000 -> fc6
5543b195f18391886b9f663d5a7eca38e84
To git@github.com:schacon/simplegit2.git
 * [new branch]          master -> master
$>
```

```
$>git remote
$>git remote add public git@github.com:schacon/simplegit2.git
$>git push public master
Counting objects: 115, done.
Compressing objects: 100% (73/73), done.
Writing objects: 100% (115/115), 11.20 KiB, done.
Total 115 (delta 42), reused 65 (delta 24)
refs/heads/master: 000000000000000000000000000000000000000000000000000000000000000 -> fc6
5543b195f18391886b9f663d5a7eca38e84
To git@github.com:schacon/simplegit2.git
 * [new branch]          master -> master
$>git remote
public
$>
```

```
$>git remote
$>git remote add public git@github.com:schacon/simplegit2.git
$>git push public master
Counting objects: 115, done.
Compressing objects: 100% (73/73), done.
Writing objects: 100% (115/115), 11.20 KiB, done.
Total 115 (delta 42), reused 65 (delta 24)
refs/heads/master: 000000000000000000000000000000000000000000000000000000000000000 -> fc6
5543b195f18391886b9f663d5a7eca38e84
To git@github.com:schacon/simplegit2.git
 * [new branch]           master -> master

$>git remote
public
$>git remote show public
* remote public
  URL: git@github.com:schacon/simplegit2.git
  Tracked remote branches
    master
$>_
```



git repository hosting



schacon

[account](#) | [profile](#) | [guides](#) | [log out](#)[repositories: all](#) | [search](#)

Source Browser

Commits

Wiki

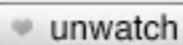
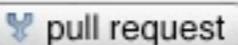
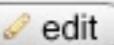
Network (1)

Admin

master

all branches

all tags

schacon / **simplegit2**

this repo is viewable by everyone

Description: another repo for simplegit for demo purposes [edit](#)Homepage: [Click to edit](#) [edit](#)Public Clone URL: <git://github.com/schacon/simplegit2.git>Your Clone URL: <git@github.com:schacon/simplegit2.git>Push URL: <git@github.com:schacon/simplegit2.git>

added file_size



Magnus Chacon (author)

17 minutes ago

commit [fc62e5543b195f18391886b9f663d5a7eca38e84](#)tree [3ec275327fe6f8474fd4d044ebb1adfe04ca38db](#)parent [f40831985e265eb8cca2d0e66c11ac48ccfe170d](#)**simplegit2 /**

name	age	message	history
README	about 14 hours ago	updated README [Magnus Chacon]	
Rakefile	39 minutes ago	changes [Magnus Chacon]	
TODO	39 minutes ago	changes [Magnus Chacon]	
lib/	16 minutes ago	added file_size [Magnus Chacon]	



git repository hosting



schacon

[account](#) | [profile](#) | [guides](#) | [log out](#)[repositories: all](#) | [search](#)

Source Browser

Commits

Wiki

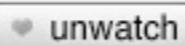
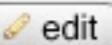
Network (1)

Admin

master

all branches

all tags

schacon / **simplegit2**

this repo is viewable by everyone

Description: another repo for simplegit for demo purposes [edit](#)Homepage: [Click to edit](#) [edit](#)Public Clone URL: <git://github.com/schacon/simplegit2.git>Your Clone URL: <git@github.com:schacon/simplegit2.git>Push URL: <git@github.com:schacon/simplegit2.git>

added file_size



Magnus Chacon (author)

17 minutes ago

commit [fc62e5543b195f18391886b9f663d5a7eca38e84](#)tree [3ec275327fe6f8474fd4d044ebb1adfe04ca38db](#)parent [f40831985e265eb8cca2d0e66c11ac48ccfe170d](#)**simplegit2 /**

name	age	message	history
README	about 14 hours ago	updated README [Magnus Chacon]	
Rakefile	39 minutes ago	changes [Magnus Chacon]	
TODO	39 minutes ago	changes [Magnus Chacon]	
lib/	16 minutes ago	added file_size [Magnus Chacon]	



schacon

✉ 0

[account](#) | [profile](#) | [guides](#) | [log out](#)

[repositories: all](#) | [search](#)

Source Browser

Commits

Wiki

Network (1)

Admin

master

all branches

all tags

[schacon / simplegit2](#)

edit

pull request

unwatch

download

this repo is viewable by everyone

Description: another repo for simplegit for demo purposes [edit](#)

Homepage: Click to edit [edit](#)

Public Clone URL: <git://github.com/schacon/simplegit2.git>

Your Clone URL: <git@github.com:schacon/simplegit2.git>

Push

add

Magnus Chacon (author)

17 minutes ago

parent [f40831985e265eb8cca2d0e66c11ac48ccfe170d](#)

git clone git://github.com/schacon/simplegit2.git

[simplegit2 /](#)

name	age	message	history
README	about 14 hours ago	updated README [Magnus Chacon]	
Rakefile	39 minutes ago	changes [Magnus Chacon]	
TODO	39 minutes ago	changes [Magnus Chacon]	
lib/	16 minutes ago	added file_size [Magnus Chacon]	

distributed workflow

distributed workflow

fetch, pull and push

fetch

```
$>git remote  
github  
$>
```

```
$>git remote  
github  
$>git remote show github  
* remote github  
  URL: git://github.com/schacon/simplegit.git  
  New remote branches (next fetch will store in remotes/github)  
    master  
$>
```

```
$>git remote  
github  
$>git remote show github  
* remote github  
  URL: git://github.com/schacon/simplegit.git  
  New remote branches (next fetch will store in remotes/github)  
    master  
$>
```

```
$>git remote  
github  
$>git remote show github  
* remote github  
  URL: git://github.com/schacon/simplegit.git  
  New remote branches (next fetch will store in remotes/github)  
    master  
$>git fetch github  
remote: Generating pack...  
remote: Done counting 106 objects.  
remote: Result has 101 objects.  
remote: Deltifying 101 objects...  
remote: 100% (101/101) done  
remote: Total 101 (delta 35), reused 101 (delta 35)  
Receiving objects: 100% (101/101), 9.49 KiB, done.  
Resolving deltas: 100% (35/35), done.  
From git://github.com/schacon/simplegit  
 * [new branch]      master      -> github/master  
$>
```

```
$>git remote  
github  
$>git remote show github  
* remote github  
  URL: git://github.com/schacon/simplegit.git  
  New remote branches (next fetch will store in remotes/github)  
    master  
$>git fetch github  
remote: Generating pack...  
remote: Done counting 106 objects.  
remote: Result has 101 objects.  
remote: Deltifying 101 objects...  
remote: 100% (101/101) done  
remote: Total 101 (delta 35), reused 101 (delta 35)  
Receiving objects: 100% (101/101), 9.49 KiB, done.  
Resolving deltas: 100% (35/35), done.  
From git://github.com/schacon/simplegit  
 * [new branch]      master      -> github/master  
$>git branch -a  
  develop  
  master  
* story95  
  github/master  
$>_
```

```
$>git remote  
github  
$>git remote show github  
* remote github  
  URL: git://github.com/schacon/simplegit.git  
  New remote branches (next fetch will store in remotes/github)  
    master  
$>git fetch github  
remote: Generating pack...  
remote: Done counting 106 objects.  
remote: Result has 101 objects.  
remote: Deltifying 101 objects...  
remote: 100% (101/101) done  
remote: Total 101 (delta 35), reused 101 (delta 35)  
Receiving objects: 100% (101/101), 9.49 KiB, done.  
Resolving deltas: 100% (35/35), done.  
From git://github.com/schacon/simplegit  
 * [new branch]      master      -> github/master  
$>git branch -a  
  develop  
  master  
* story95  
  github/master  
$>_
```

pull

pull = fetch + merge

push

```
$>git remote  
public  
$>
```

```
$>git remote  
public  
$>git remote show public  
* remote public  
  URL: git@github.com:schacon/simplegit.git  
  Tracked remote branches  
    master  
$>
```

```
$>git remote  
public  
$>git remote show public  
* remote public  
  URL: git@github.com:schacon/simplegit.git  
Tracked remote branches  
  master  
$>
```

```
$>git remote  
public  
$>git remote show public  
* remote public  
  URL: git@github.com:schacon/simplegit.git  
  Tracked remote branches  
    master  
$>git push public master  
Counting objects: 7, done.  
Compressing objects: 100% (3/3), done.  
Writing objects: 100% (4/4), 365 bytes, done.  
Total 4 (delta 2), reused 0 (delta 0)  
refs/heads/master: f40831985e265eb8cca2d0e66c11ac48ccfe170d -> fc62  
5543b195f18391886b9f663d5a7eca38e84  
To git@github.com:schacon/simplegit.git  
  f408319..fc62e55  master -> master  
$>_
```

git push

```
$>git remote  
public  
$>git remote show public  
* remote public  
  URL: git@github.com:schacon/simplegit.git  
  Tracked remote branches  
    master  
$>git push public master  
Counting objects: 7, done.  
Compressing objects: 100% (3/3), done.  
Writing objects: 100% (4/4), 365 bytes, done.  
Total 4 (delta 2), reused 0 (delta 0)  
refs/heads/master: f40831985e265eb8cca2d0e66c11ac48ccfe170d -> fc62  
5543b195f18391886b9f663d5a7eca38e84  
To git@github.com:schacon/simplegit.git  
  f408319..fc62e55  master -> master  
$>_
```

```
git push origin branchA:branchB
```

a who in the what now?

local repo

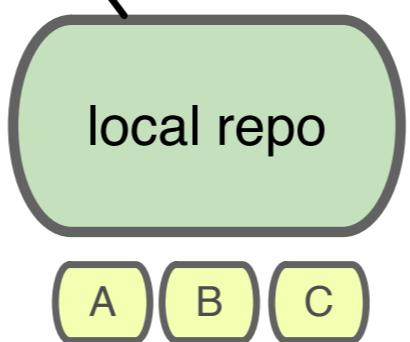
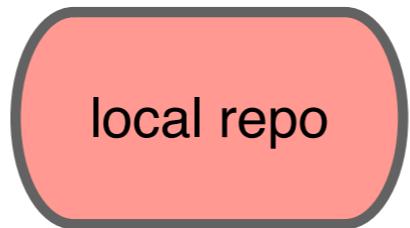
public repo

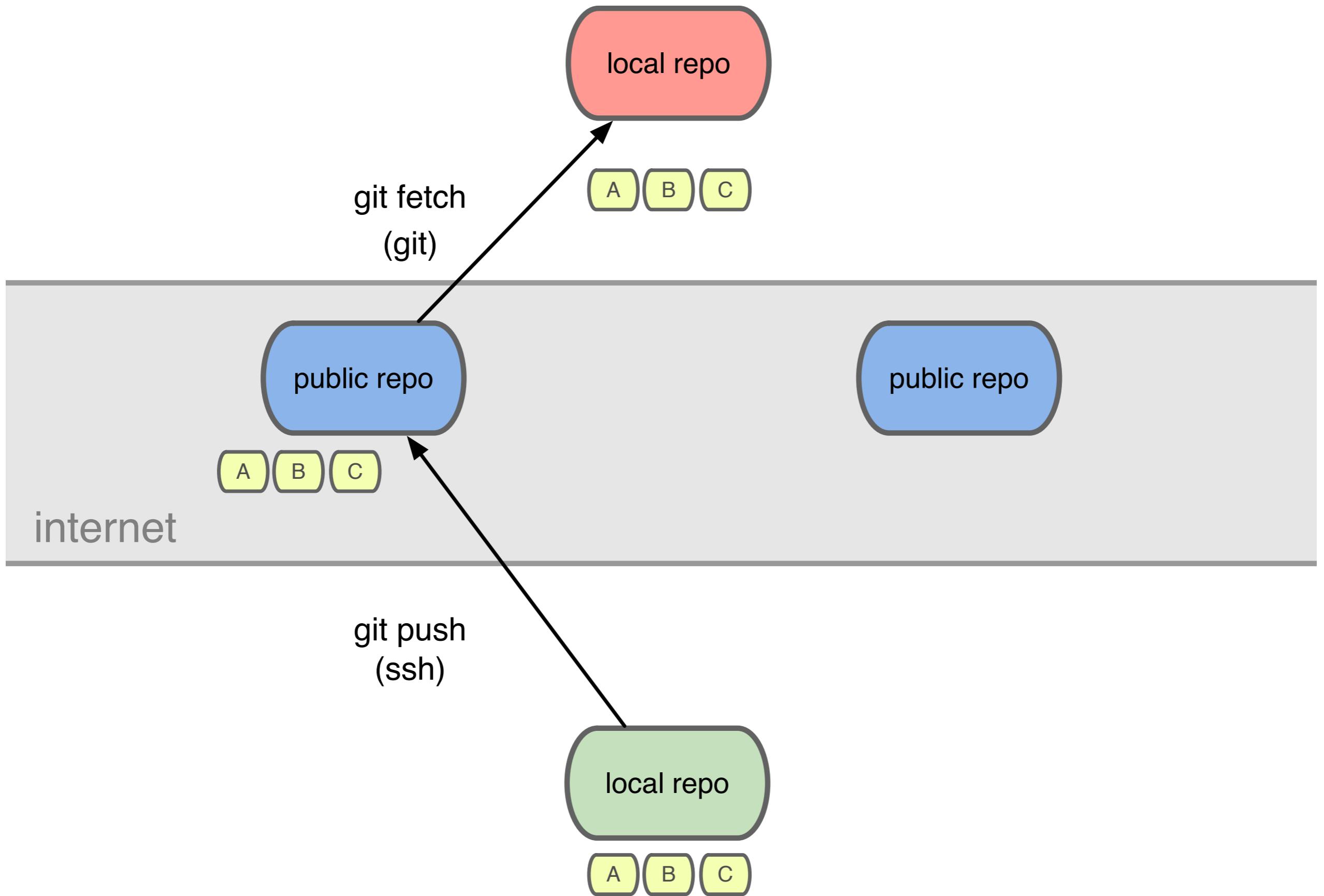
public repo

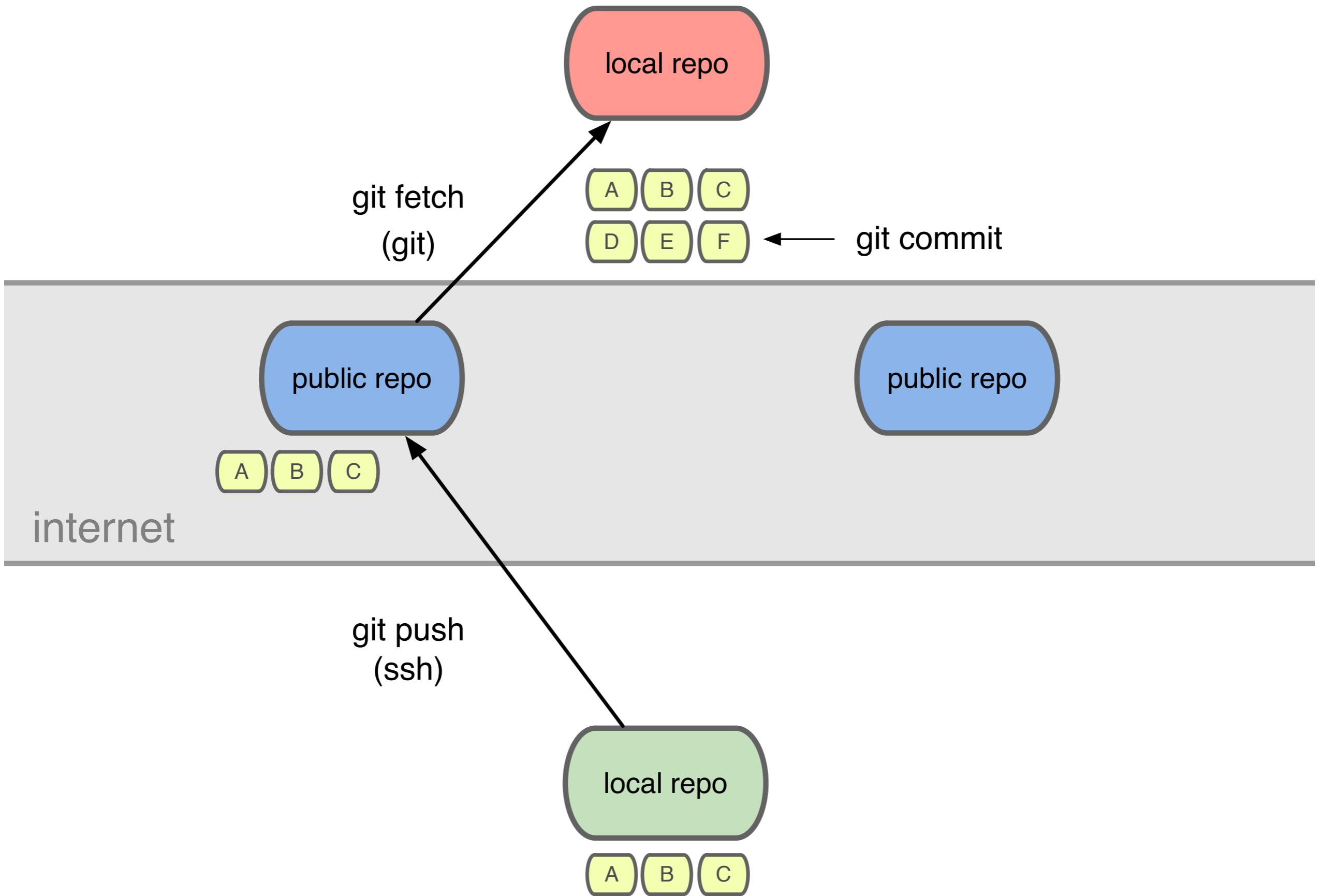
internet

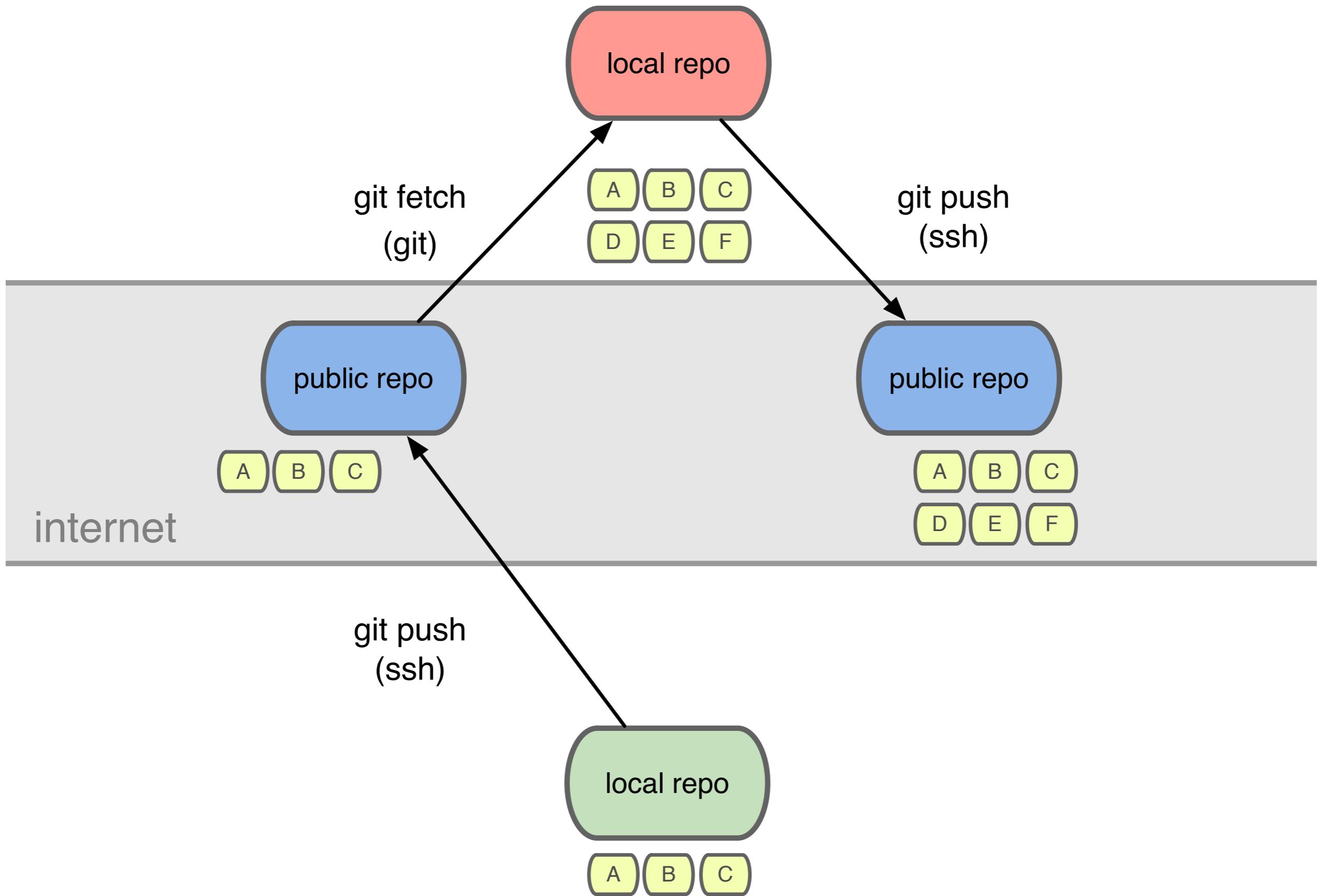
local repo

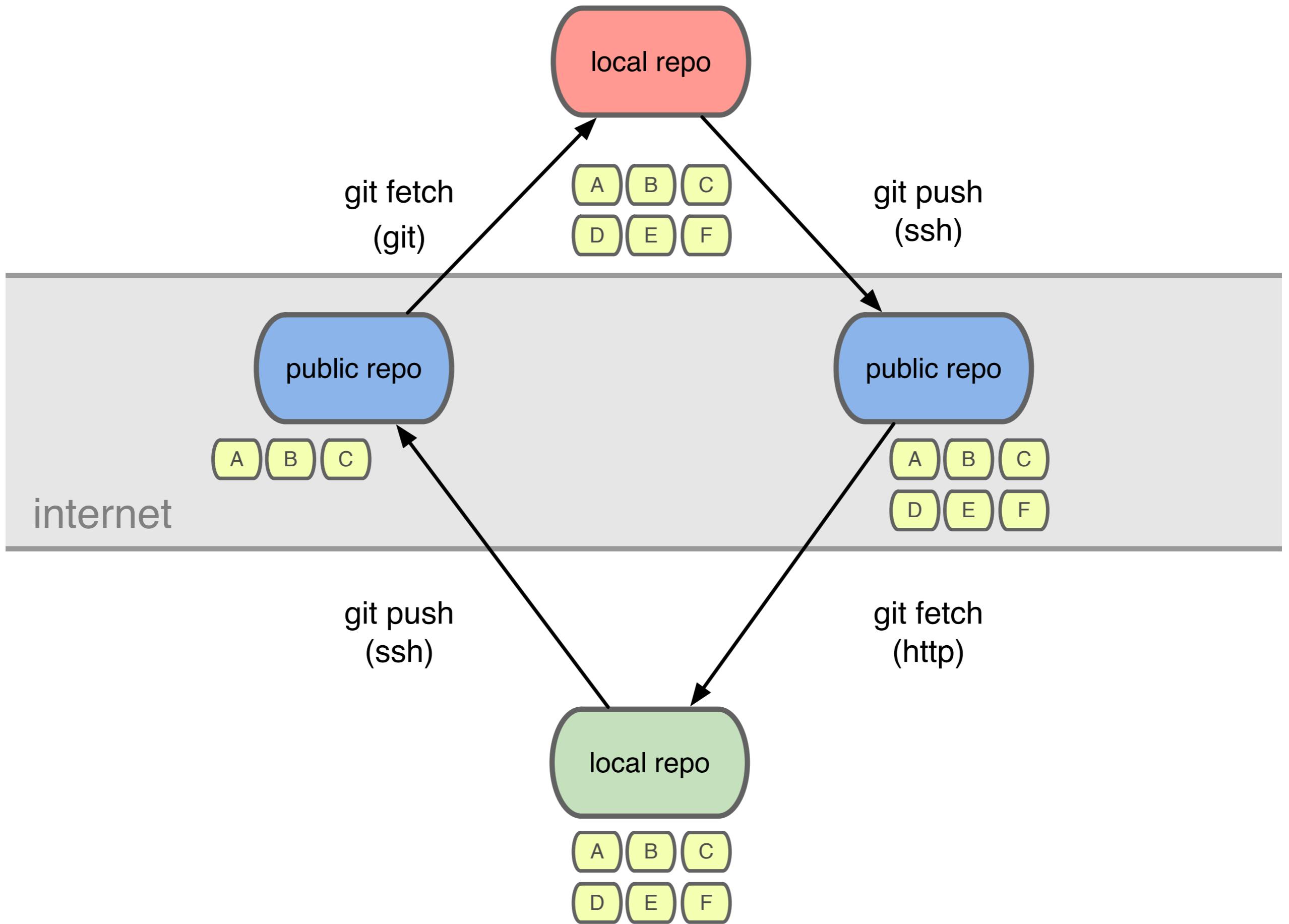
A B C

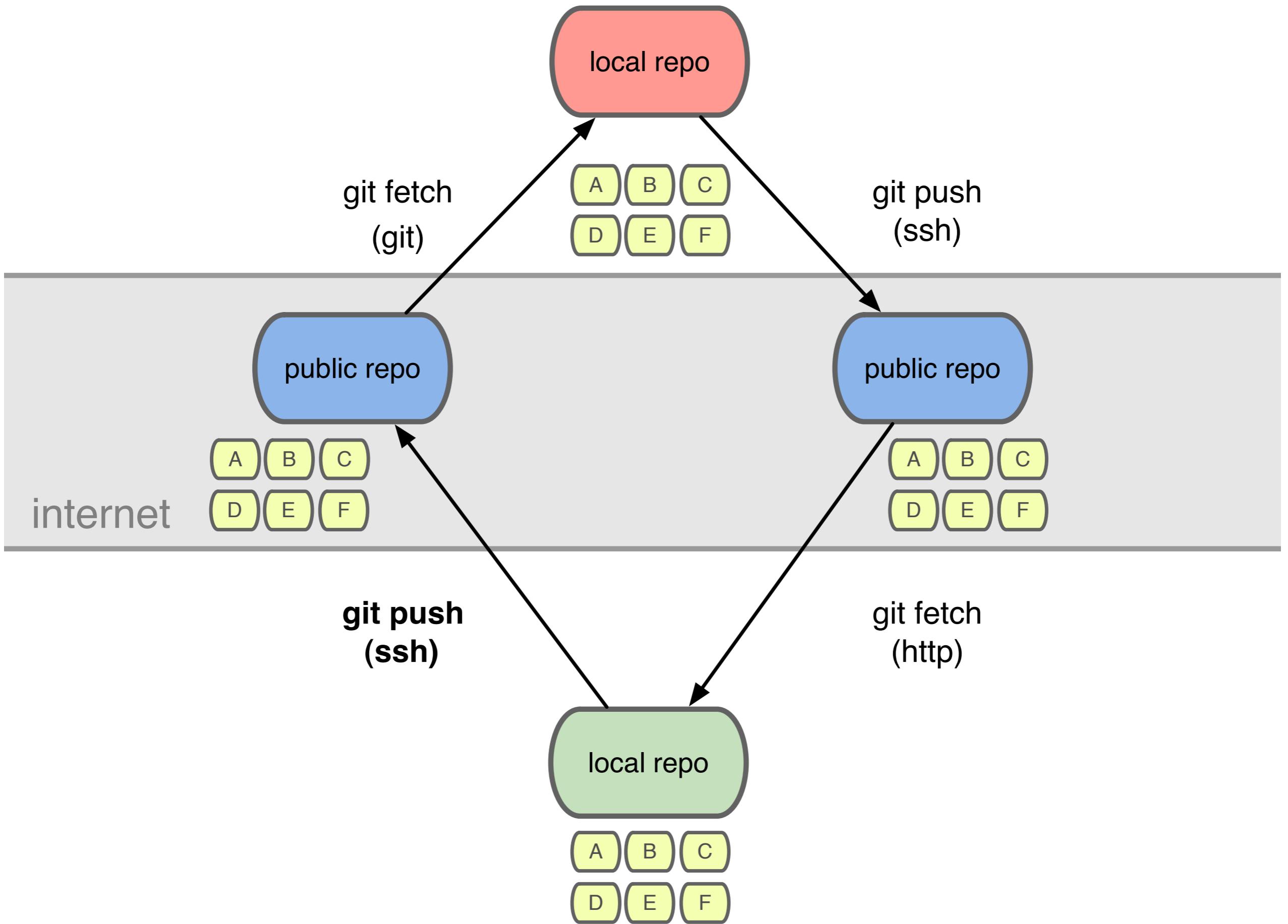








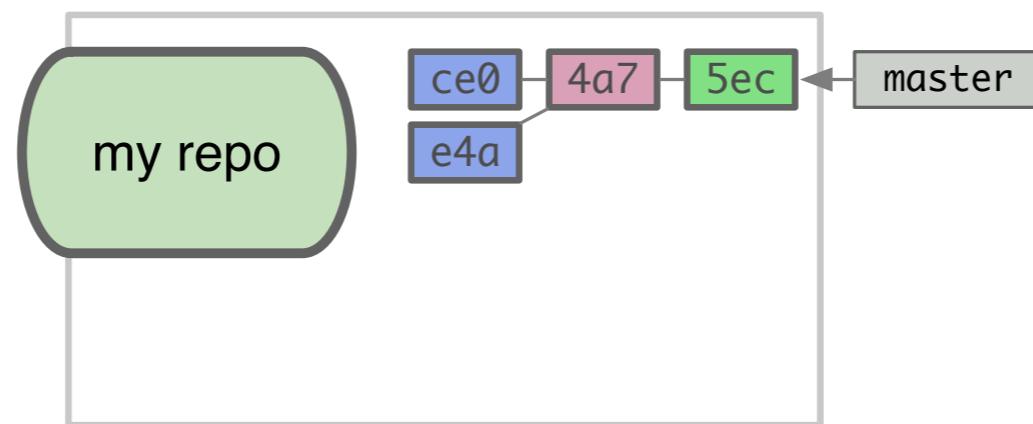




multiple remotes

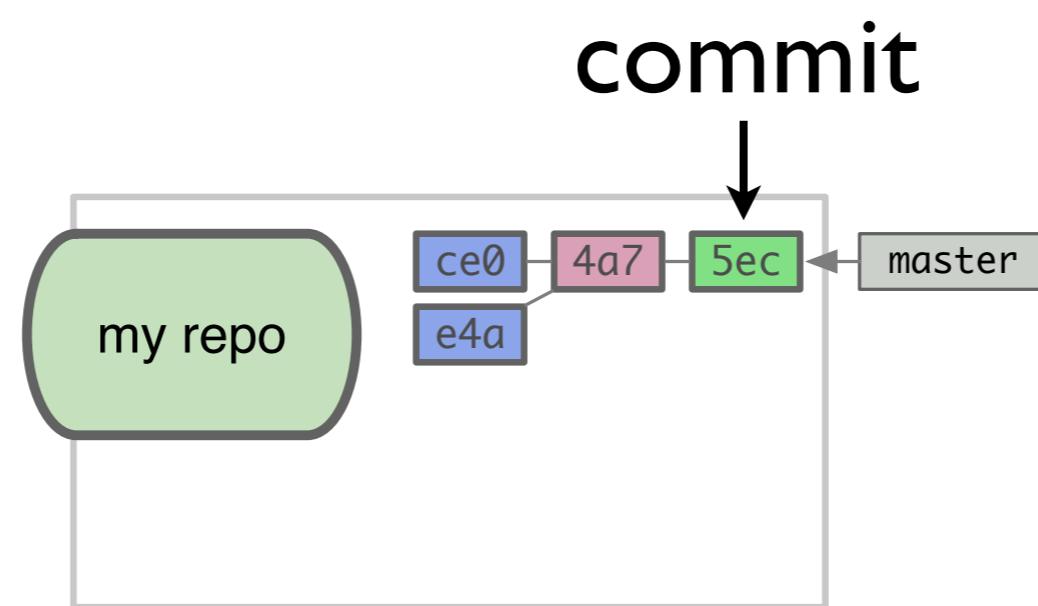
developer
nick

developer
jessica



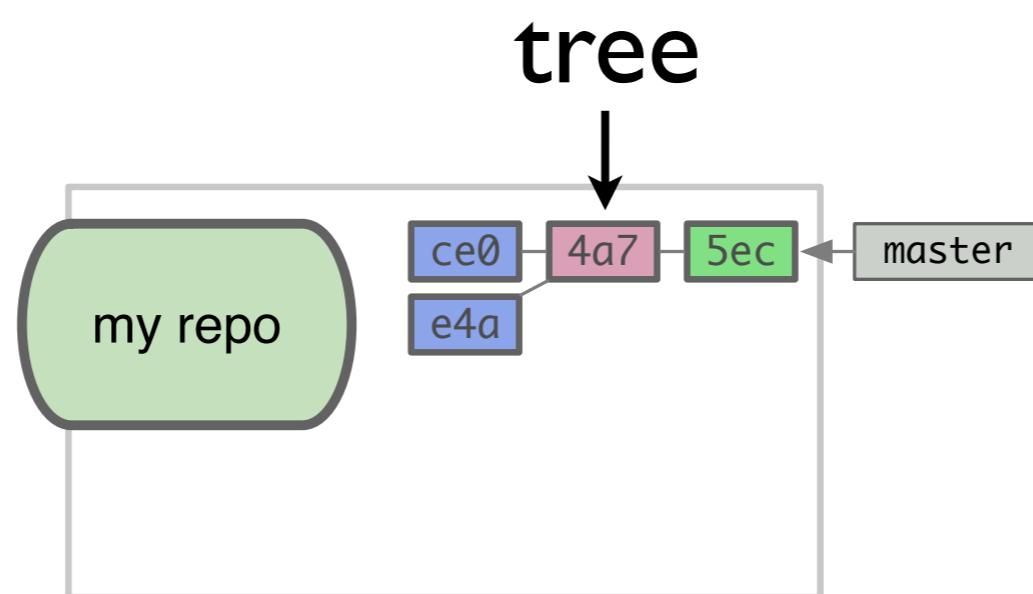
developer
nick

developer
jessica



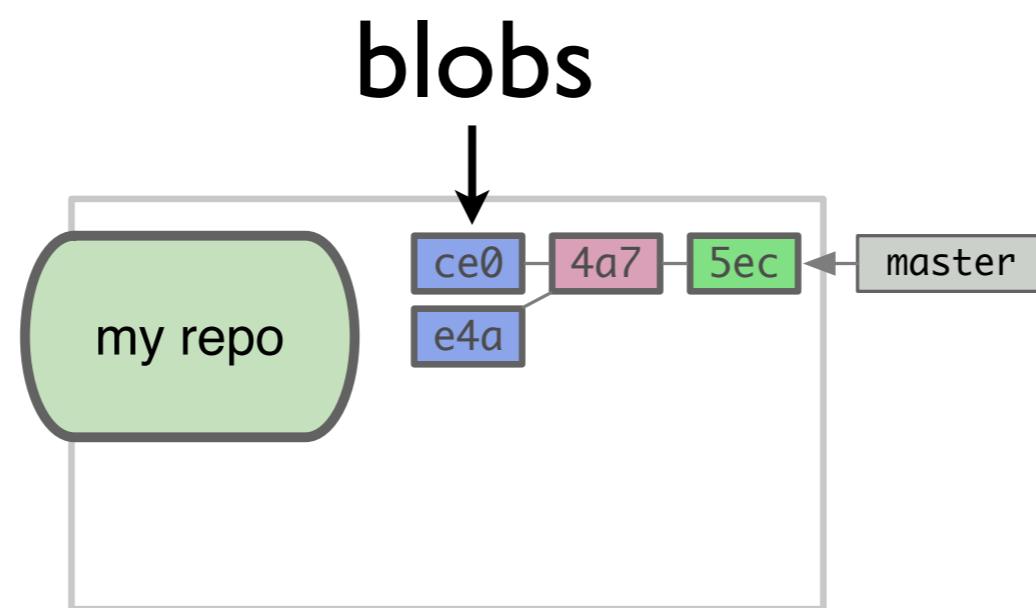
developer
nick

developer
jessica



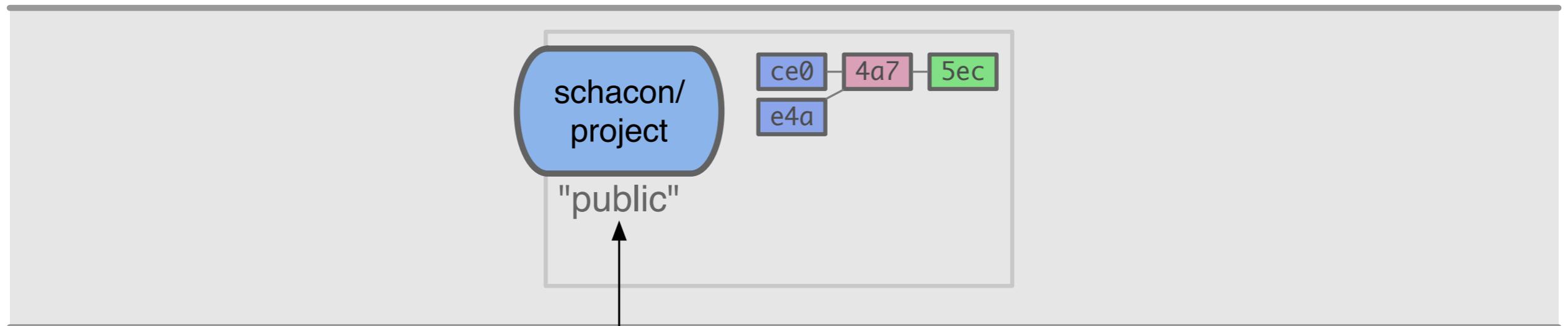
developer
nick

developer
jessica

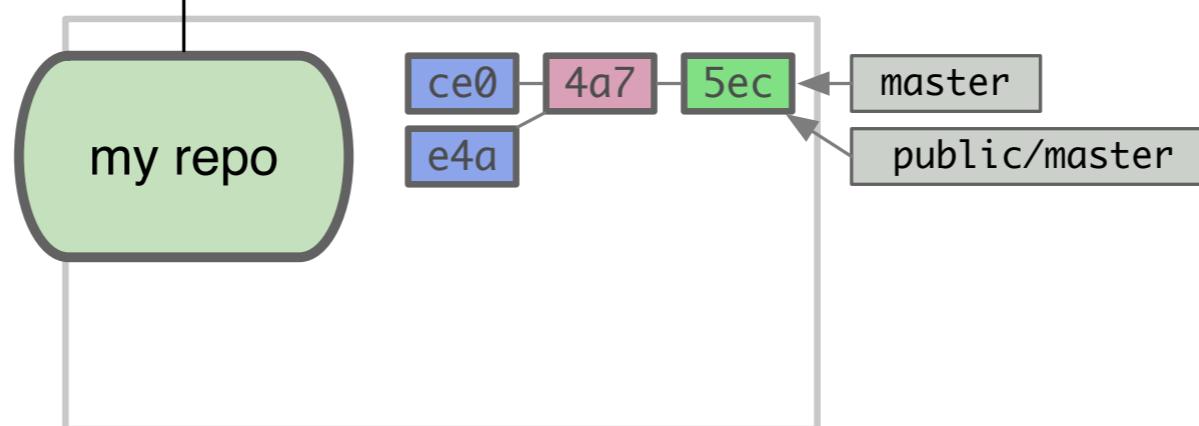


developer
nick

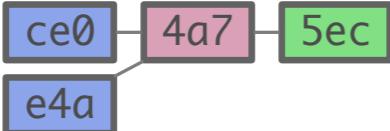
developer
jessica



git push public



developer
nick



developer
jessica

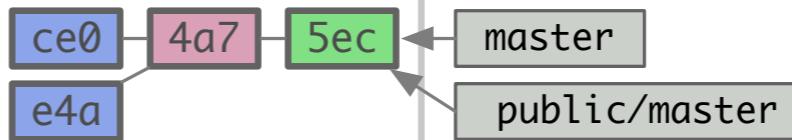
git clone (url)

schacon/
project

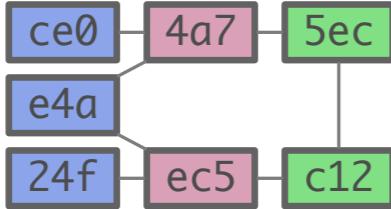
"public"



my repo



developer
nick

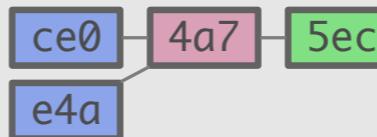


developer
jessica

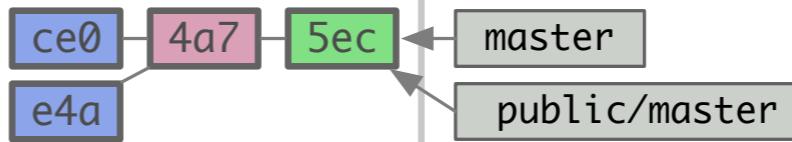
git commit

schacon/
project

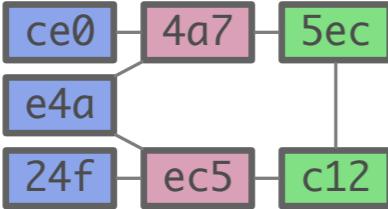
"public"



my repo

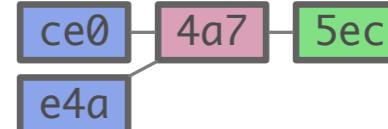


developer
nick



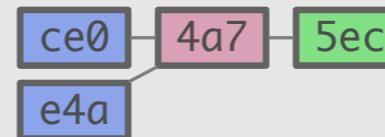
git clone (url)

developer
jessica

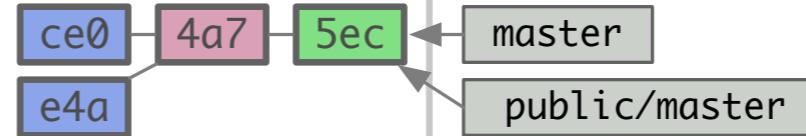


schacon/
project

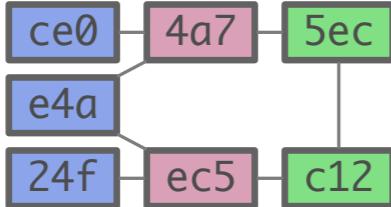
"public"



my repo

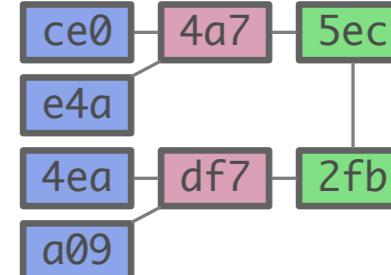


developer
nick



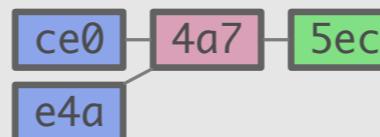
git commit

developer
jessica

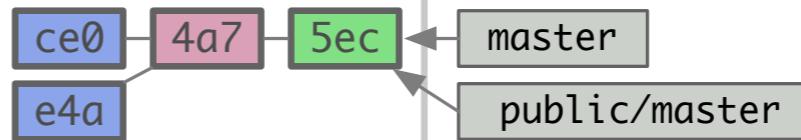


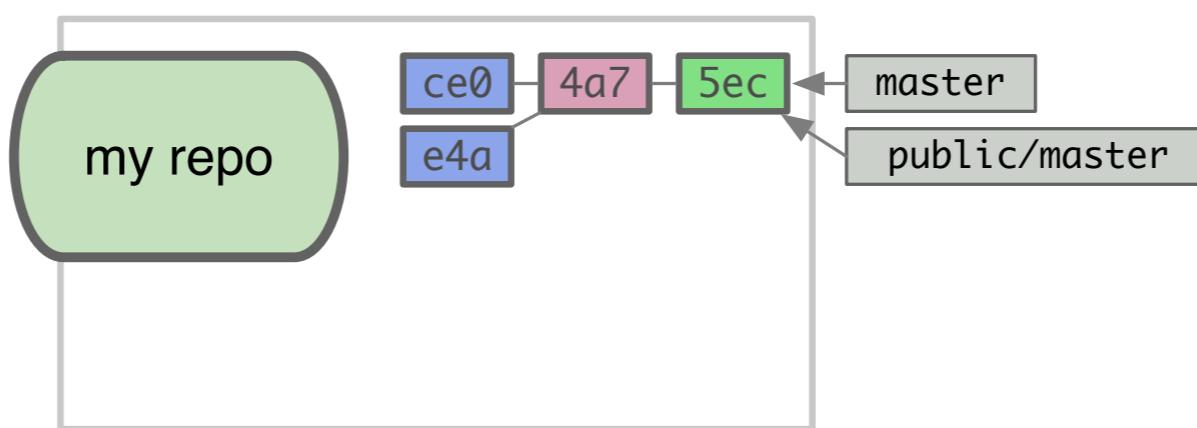
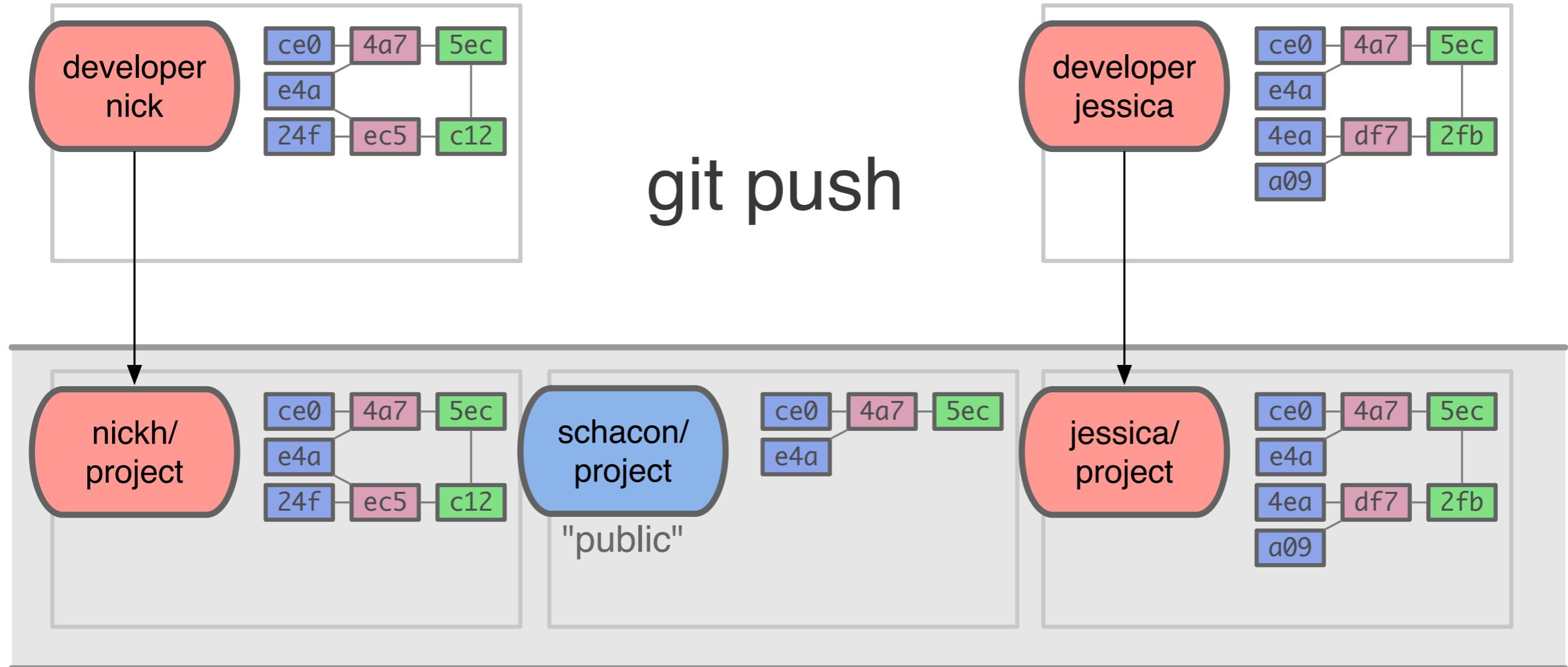
schacon/
project

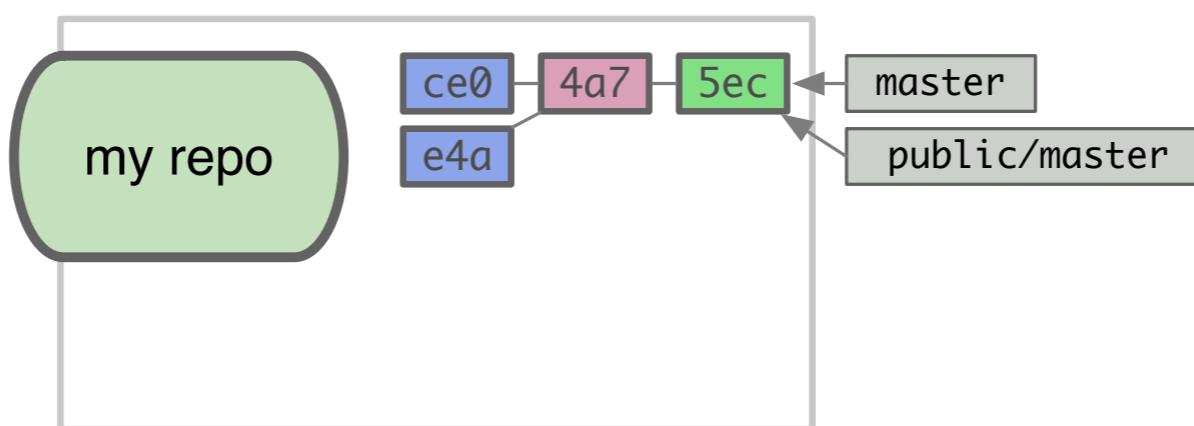
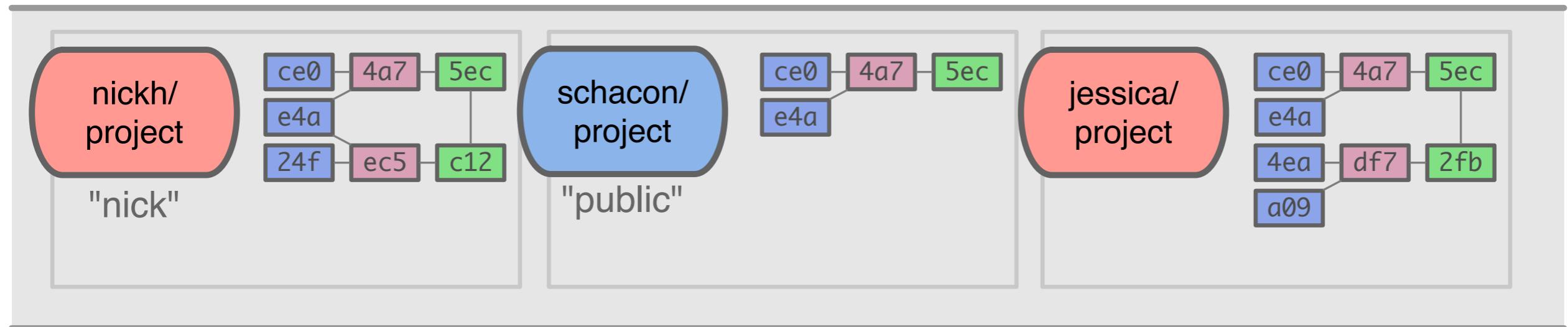
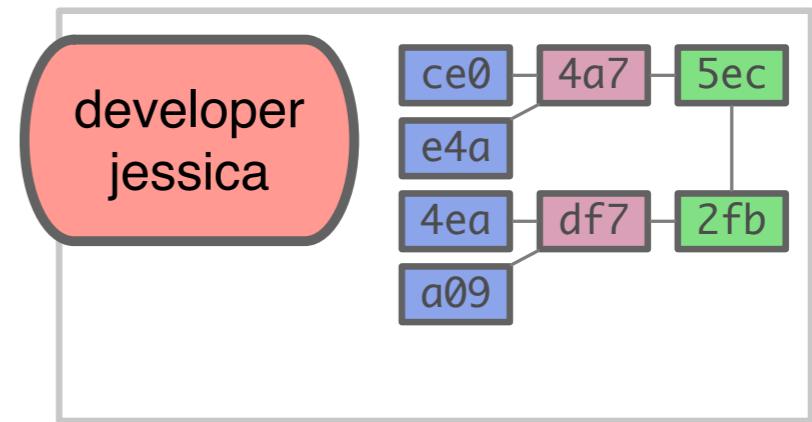
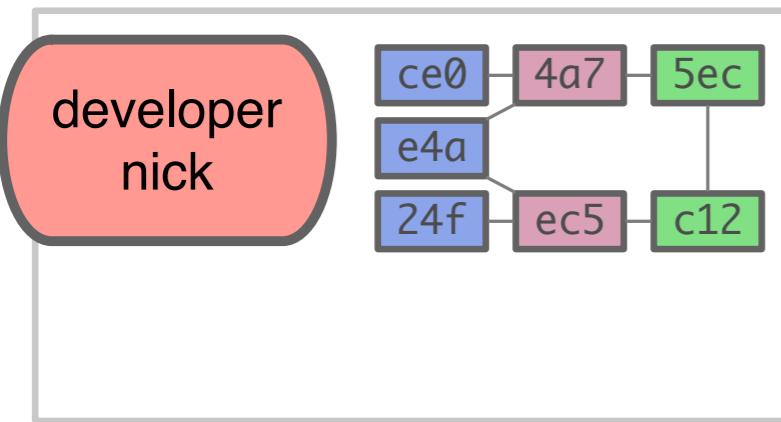
"public"



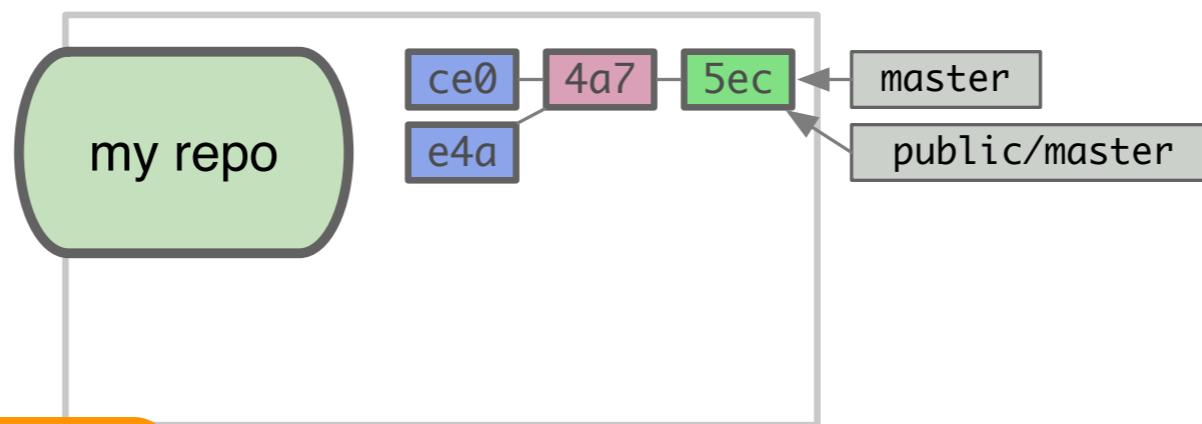
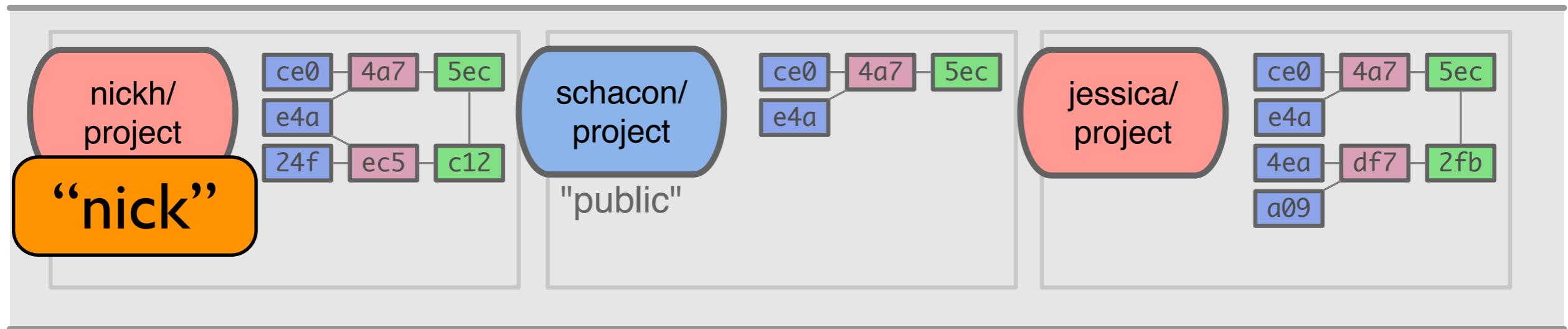
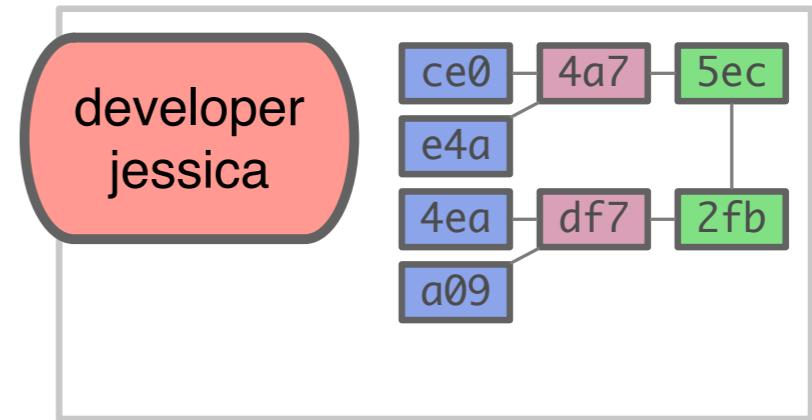
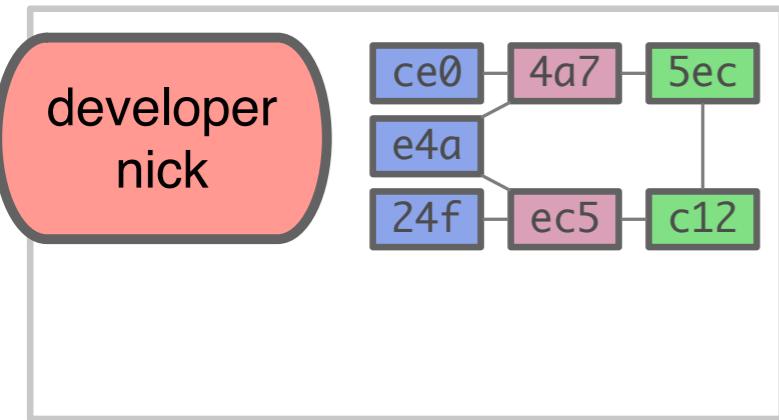
my repo



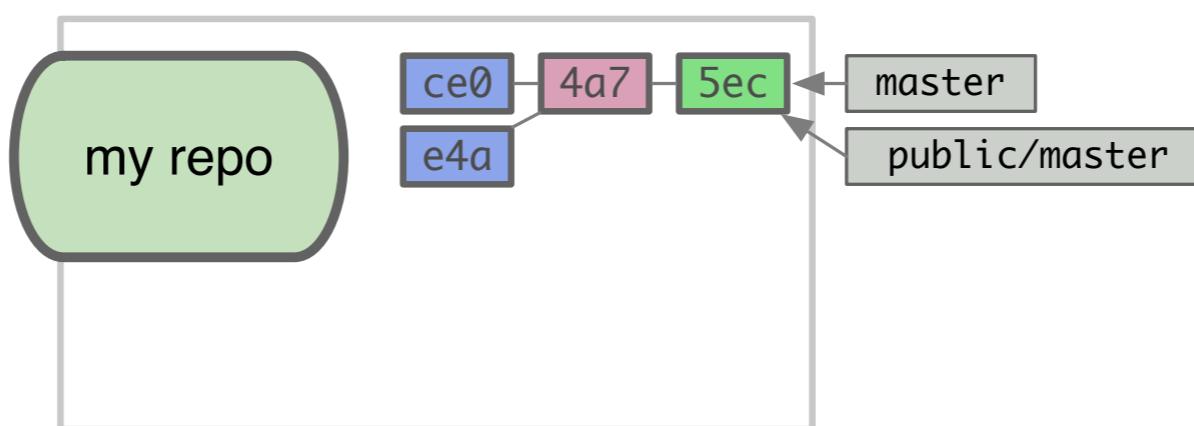
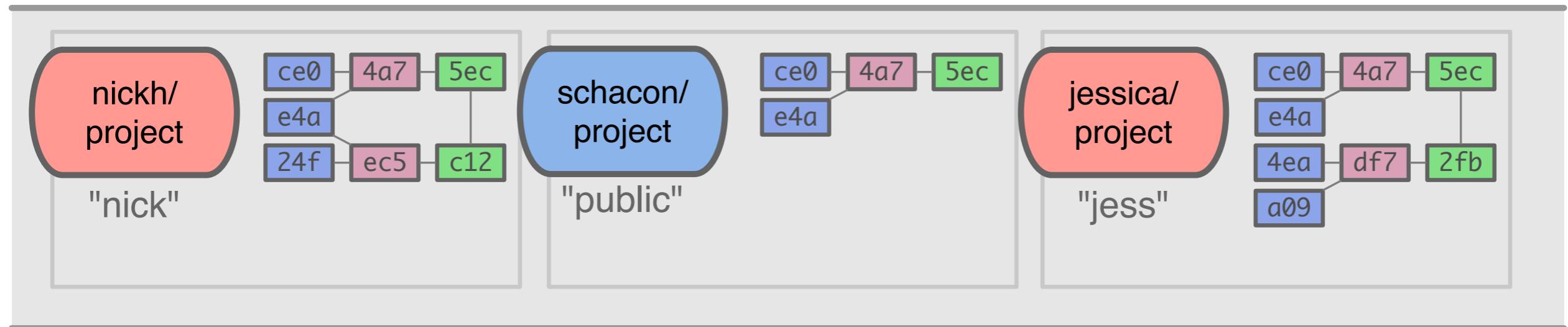
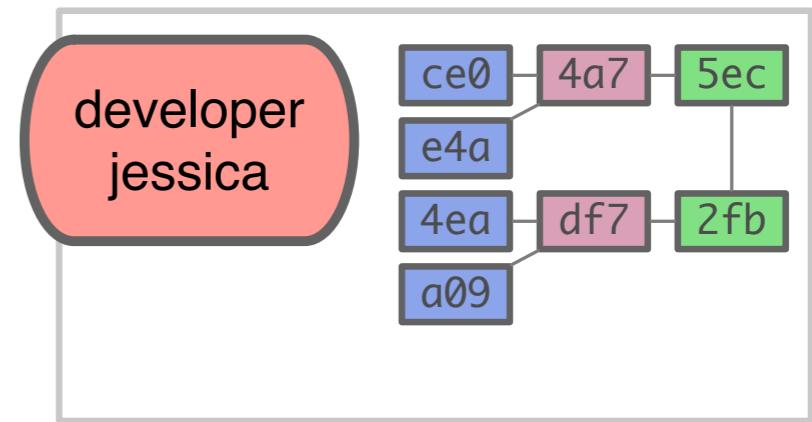
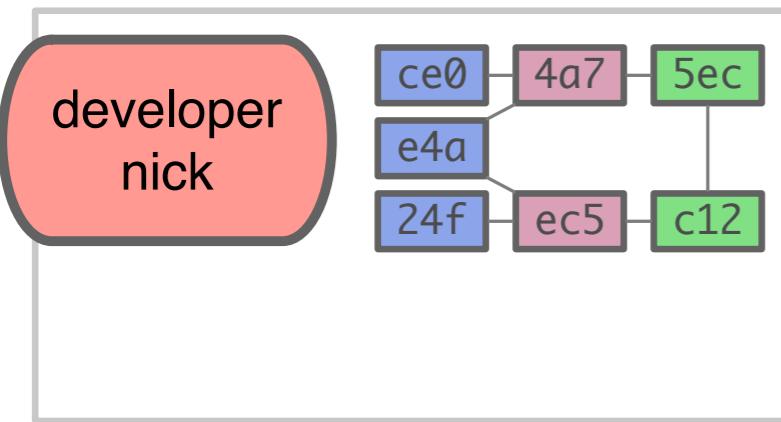




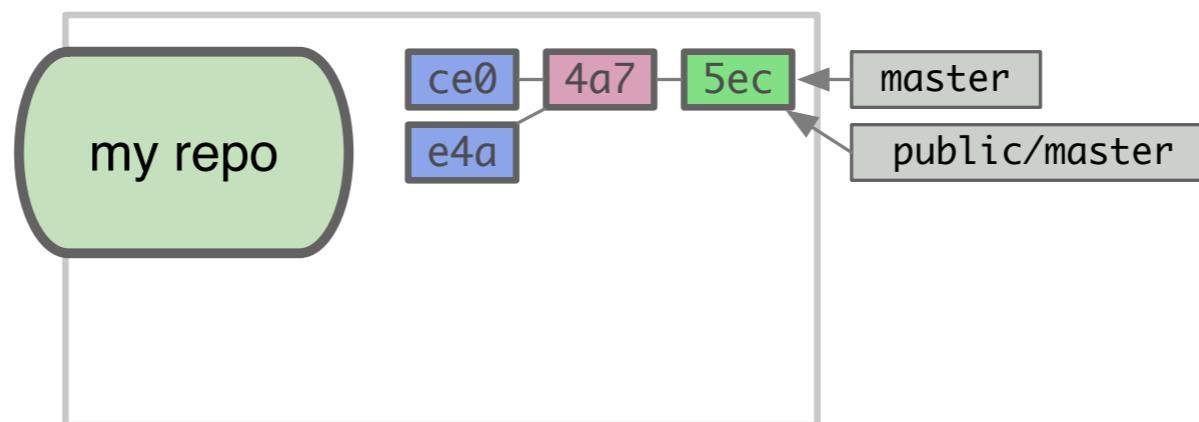
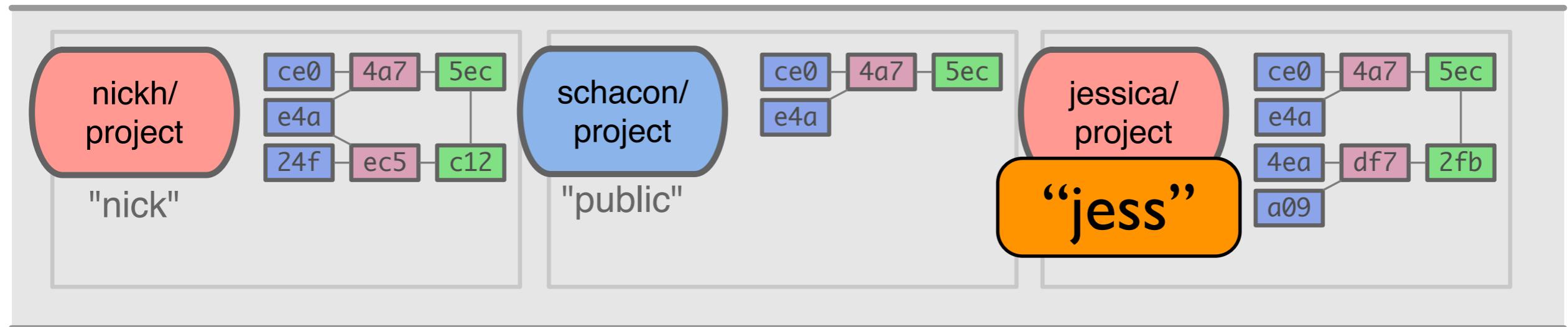
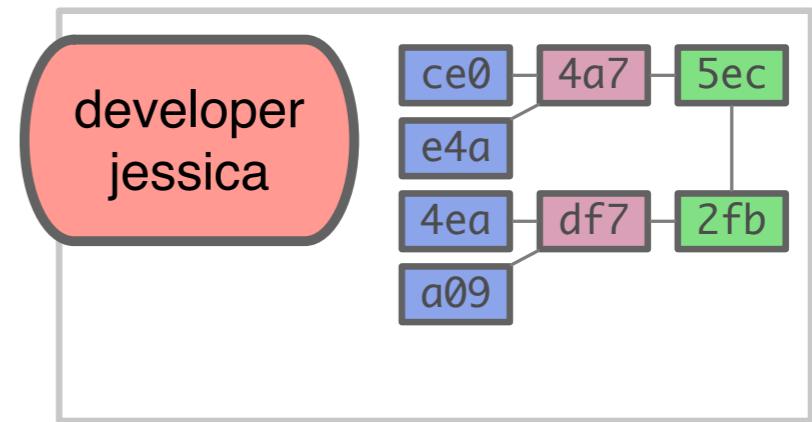
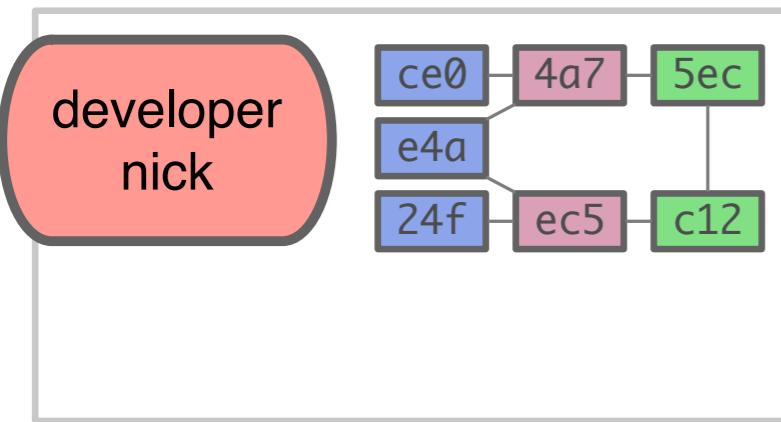
`git remote add nick git://github.com/nickh/project.git`



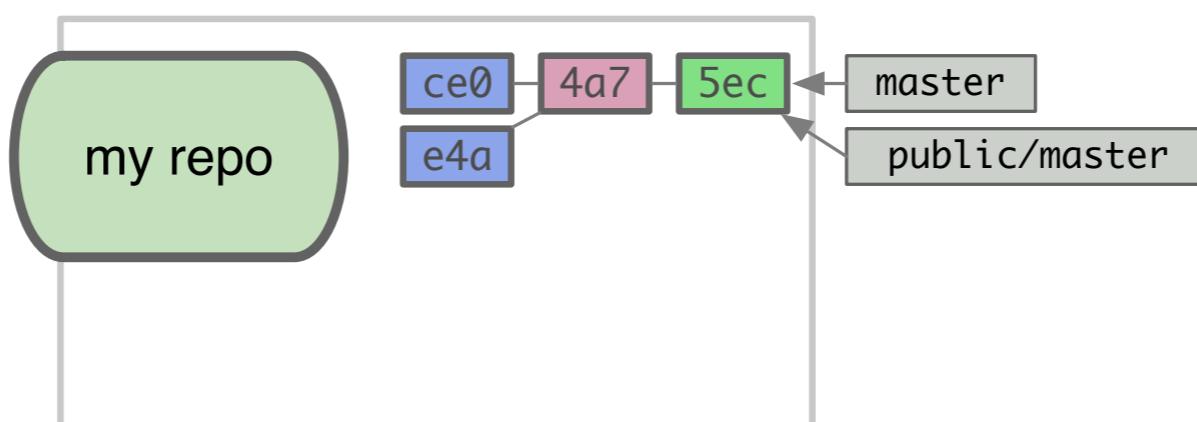
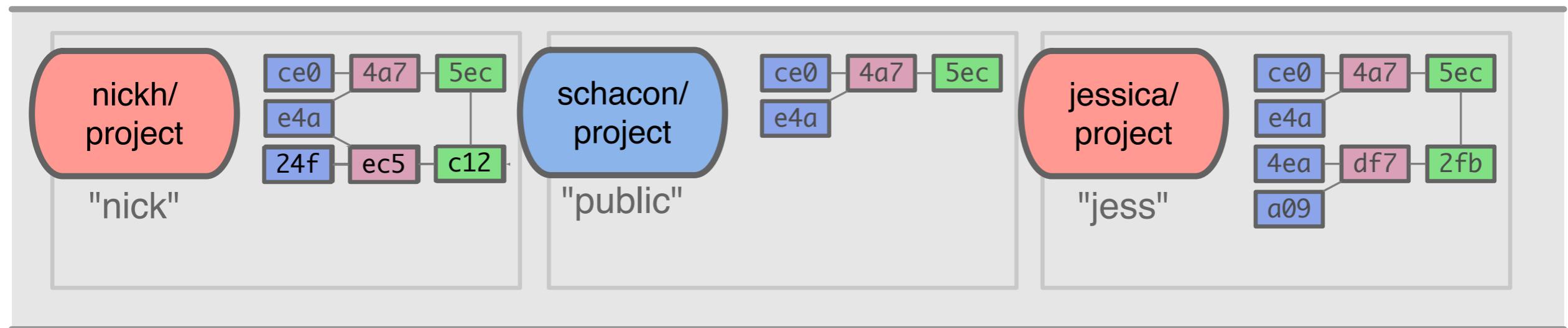
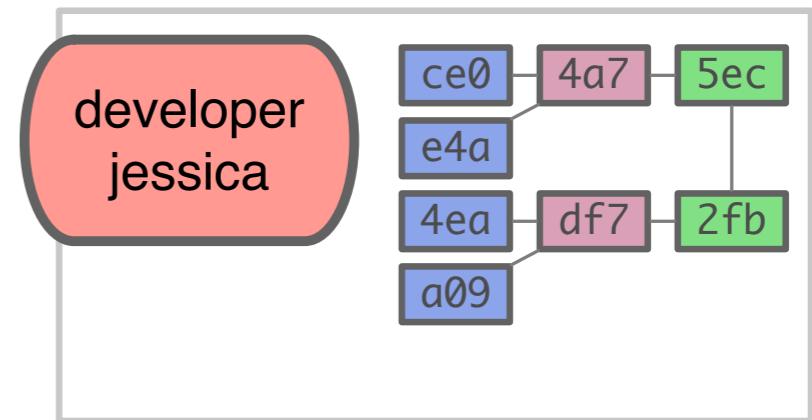
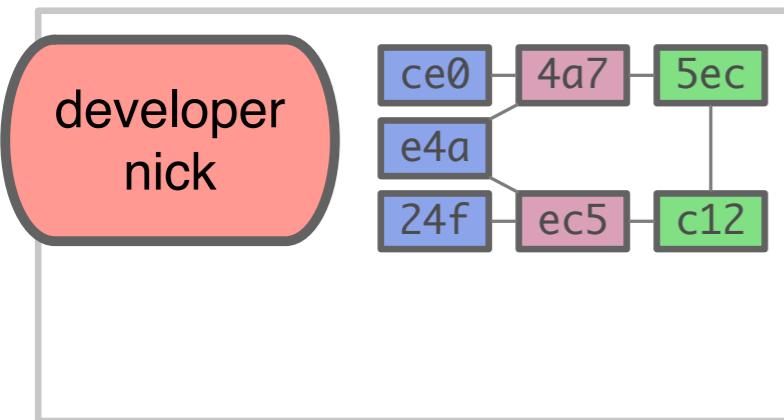
`git remote add nick git://github.com/nickh/project.git`



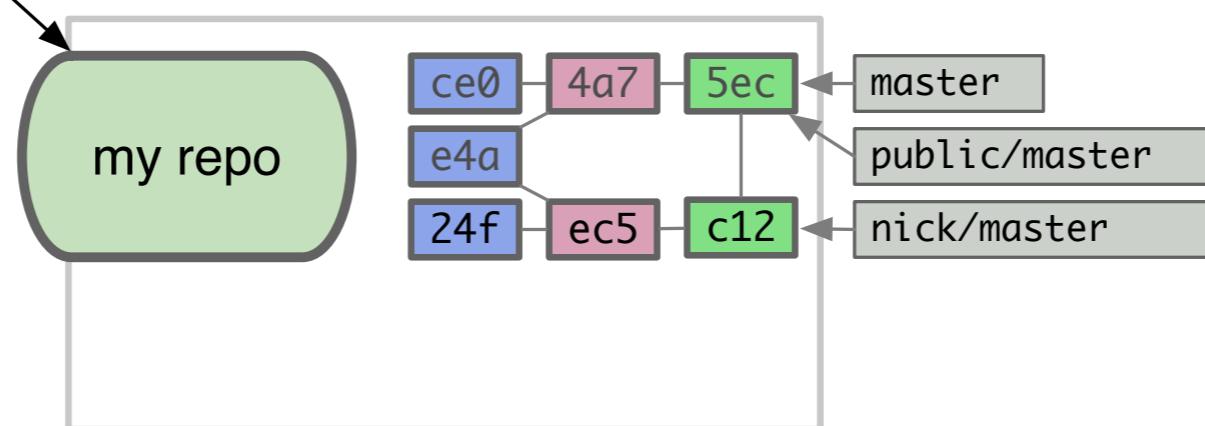
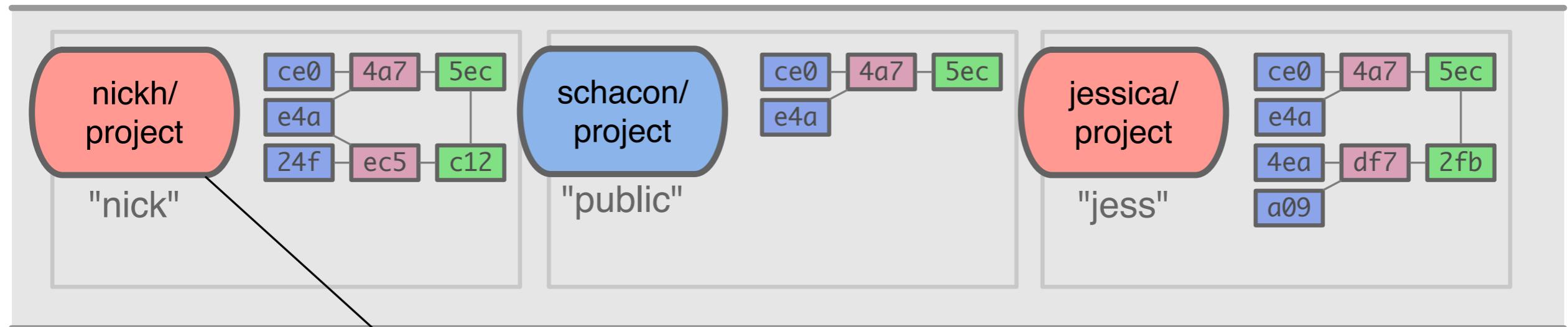
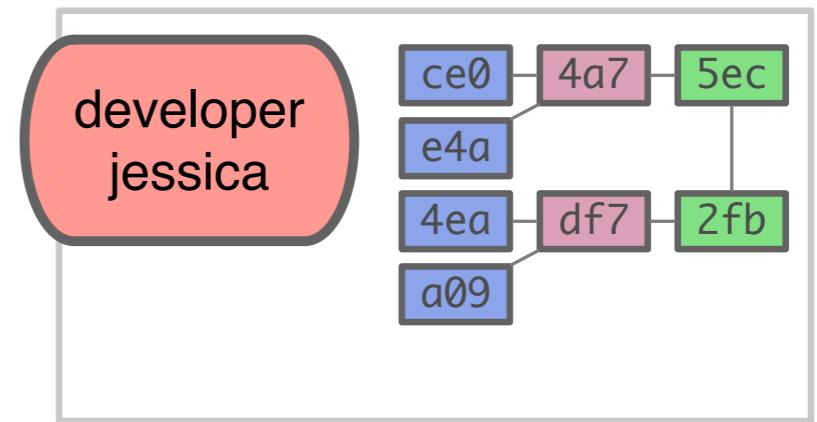
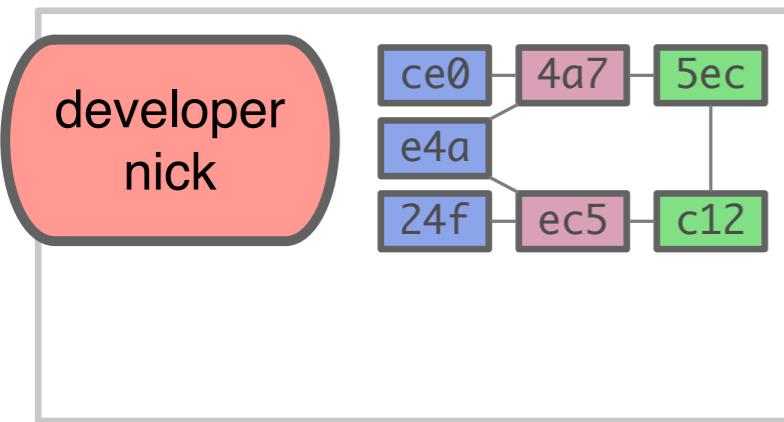
`git remote add jess git://github.com/jessica/project.git`

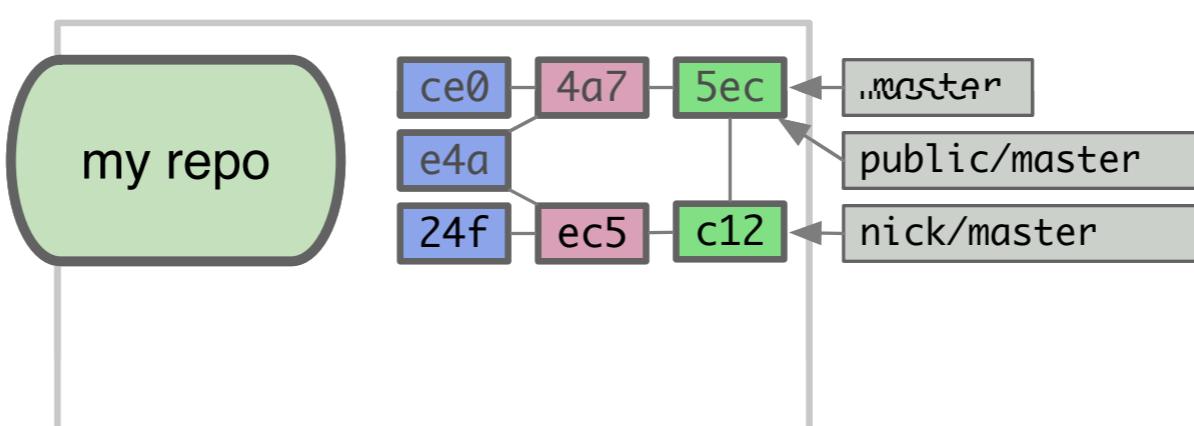
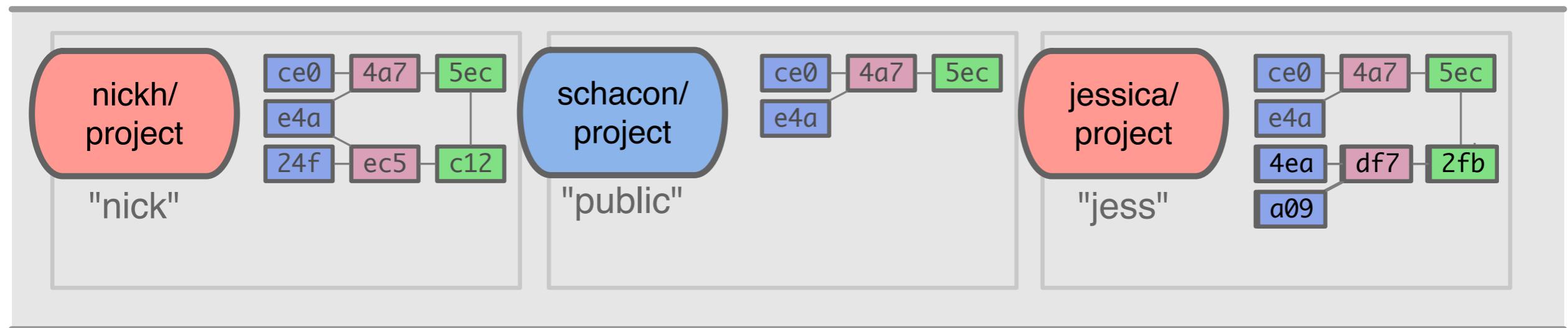
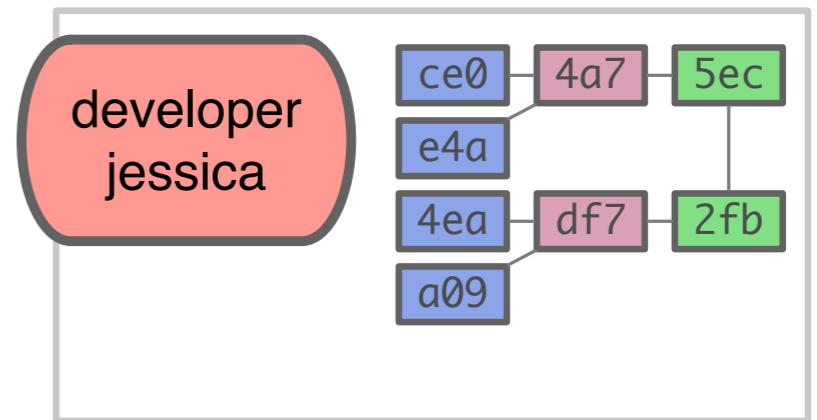
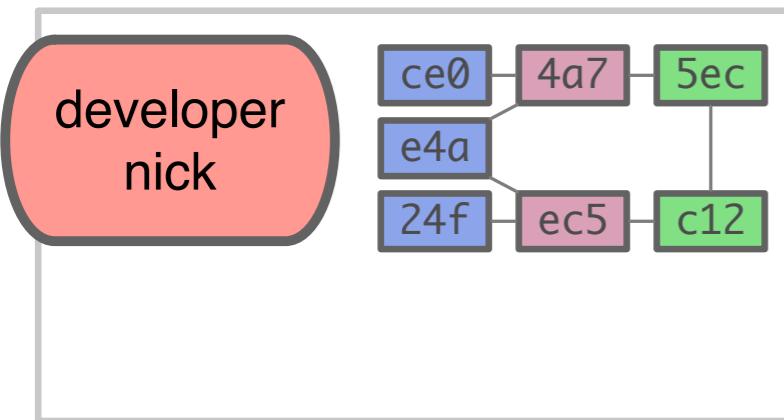


git remote add jess git://github.com/jessica/project.git

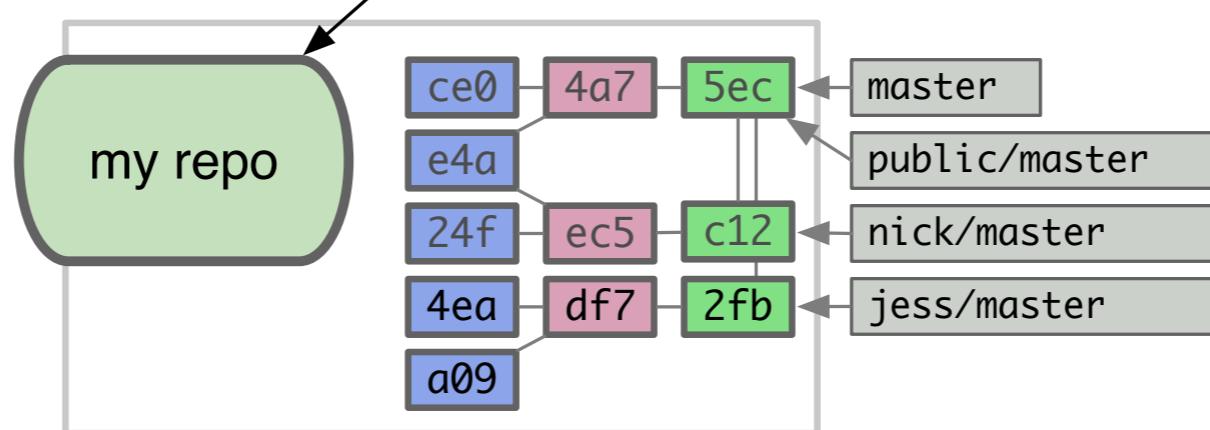
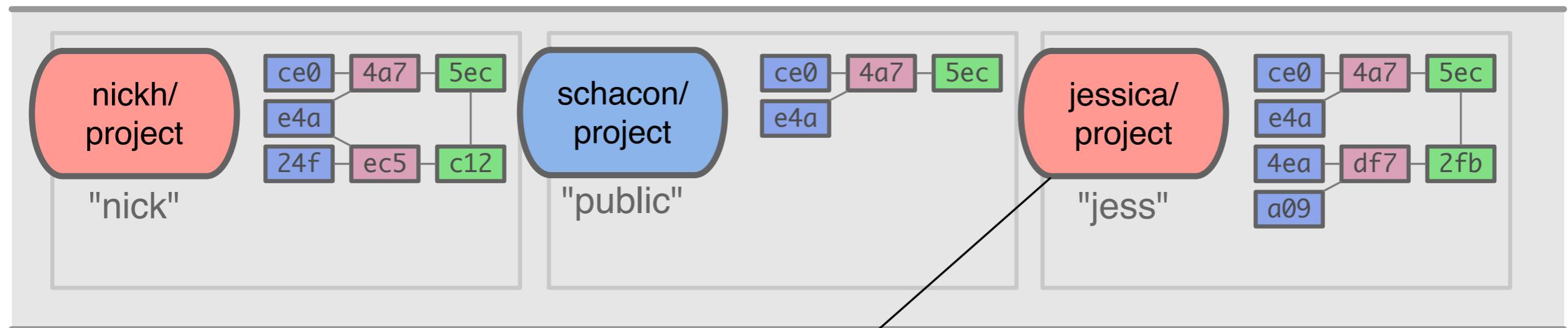
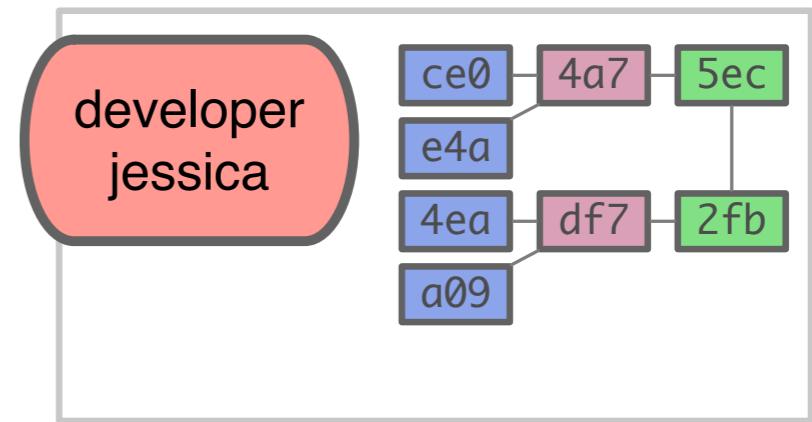
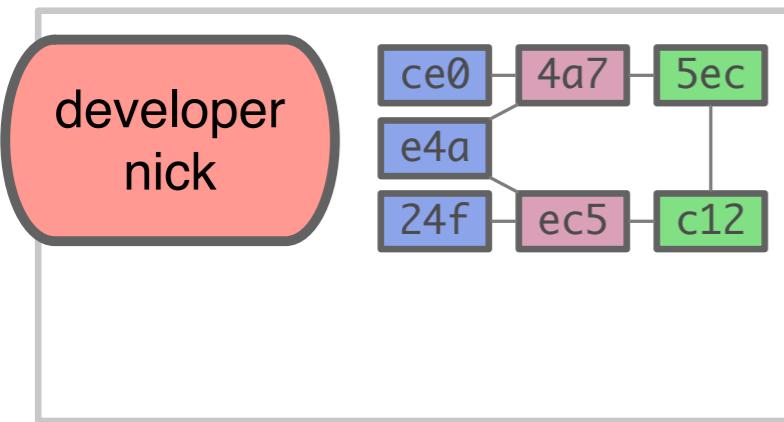


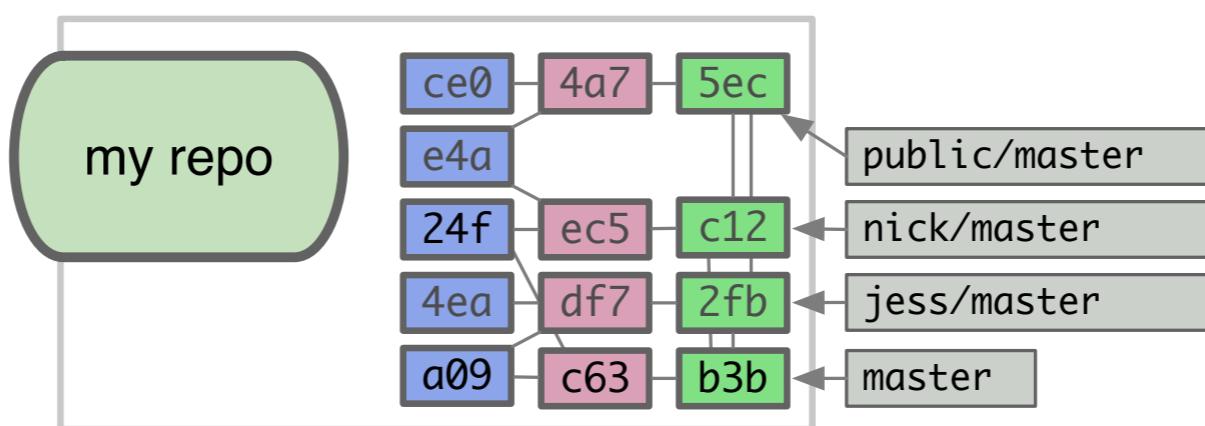
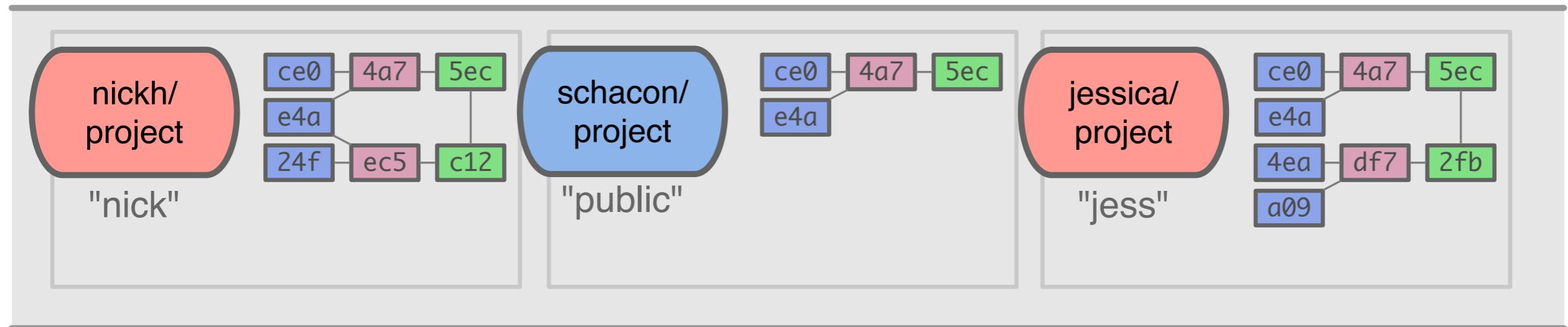
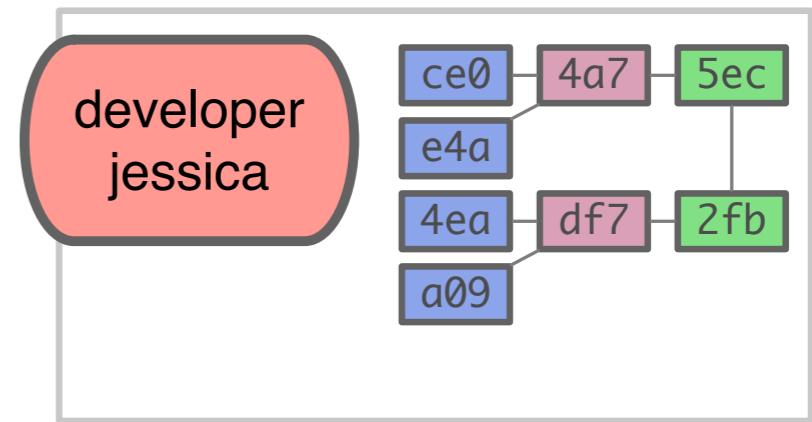
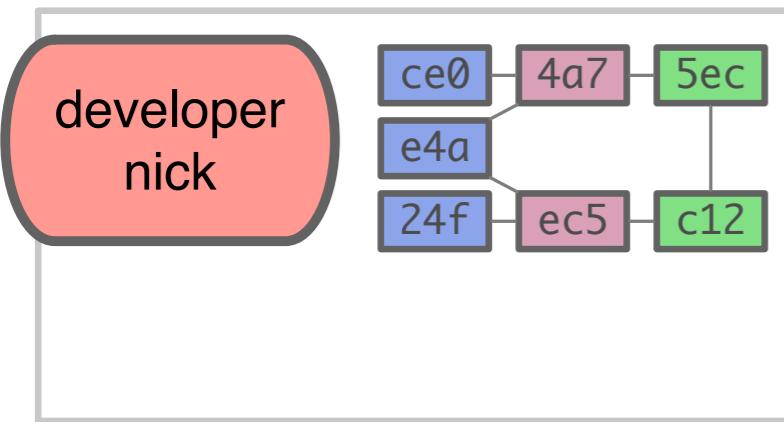
git fetch nick



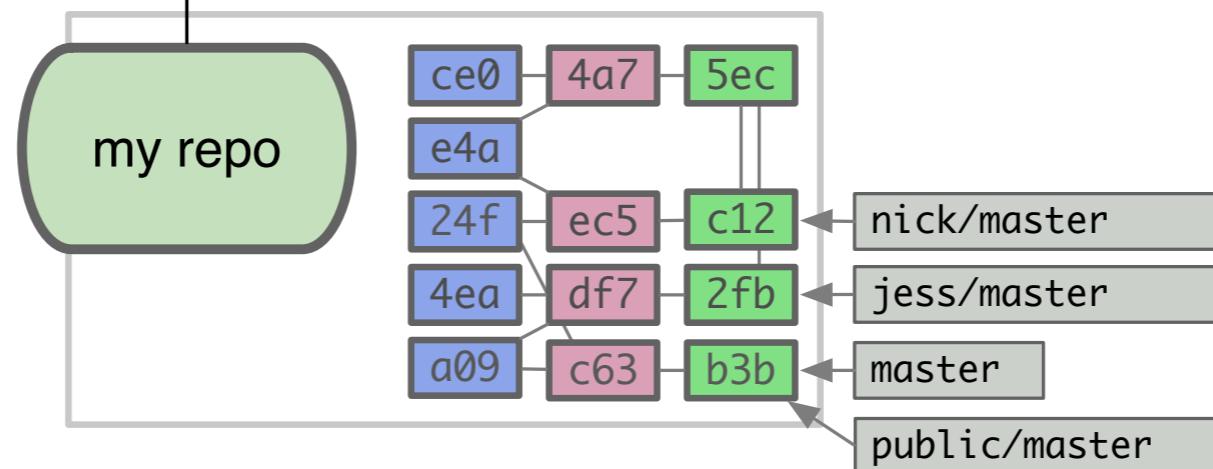
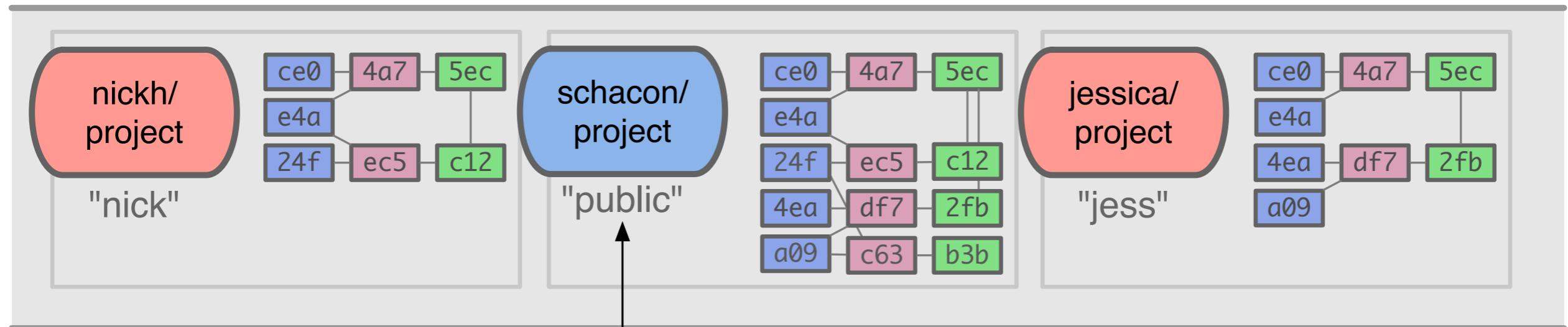
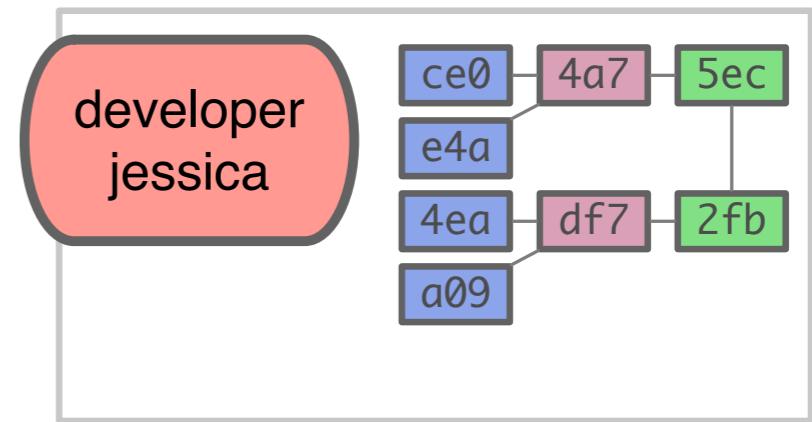
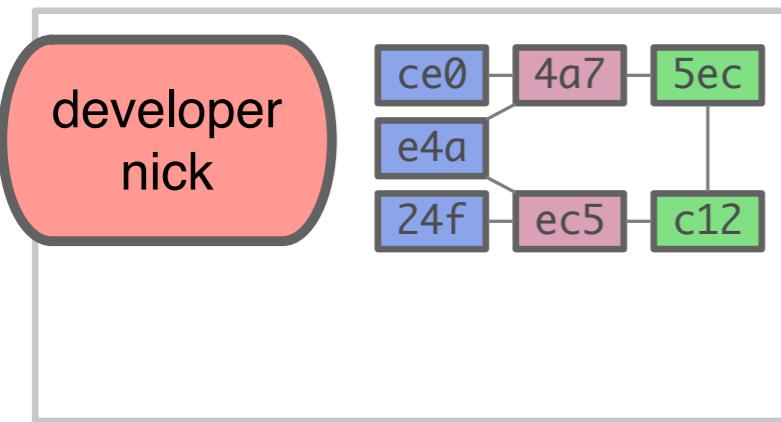


git fetch jess

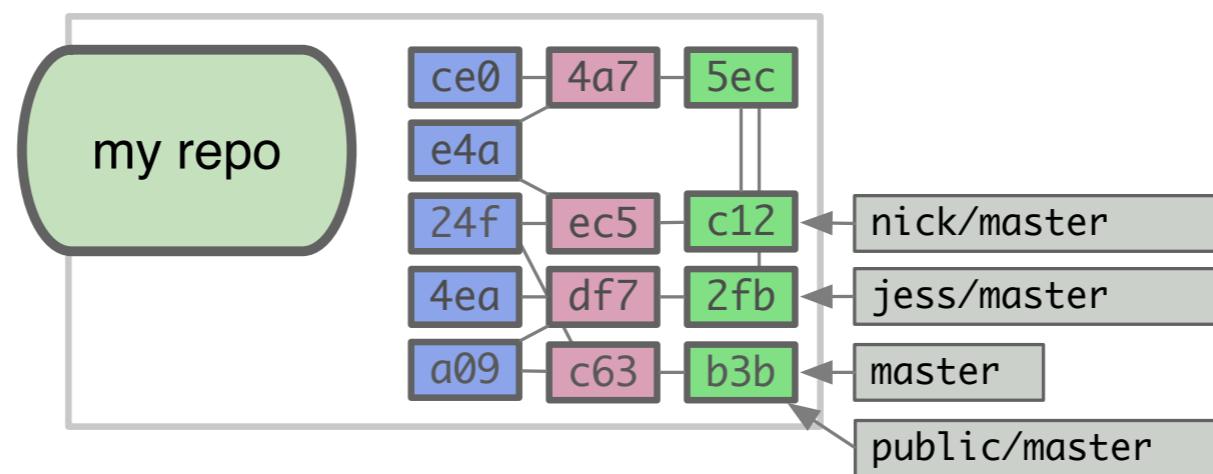
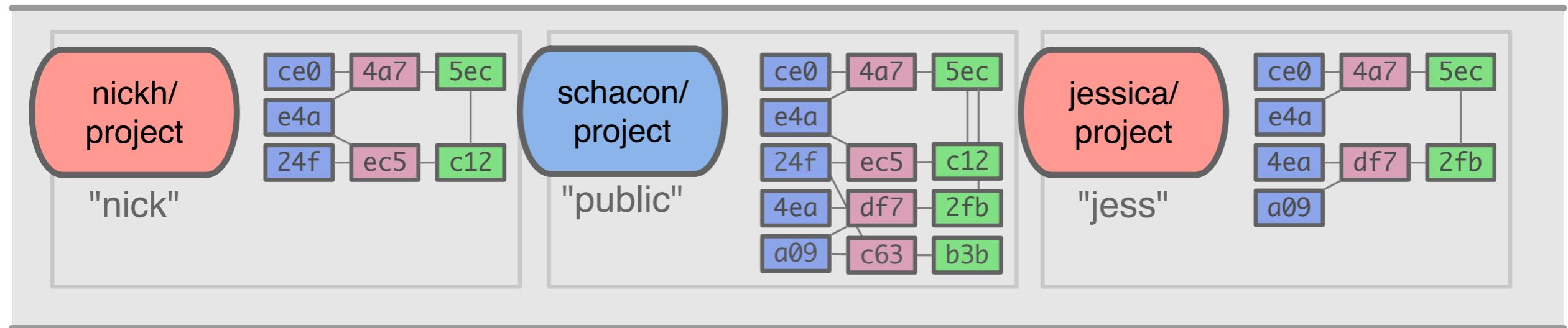
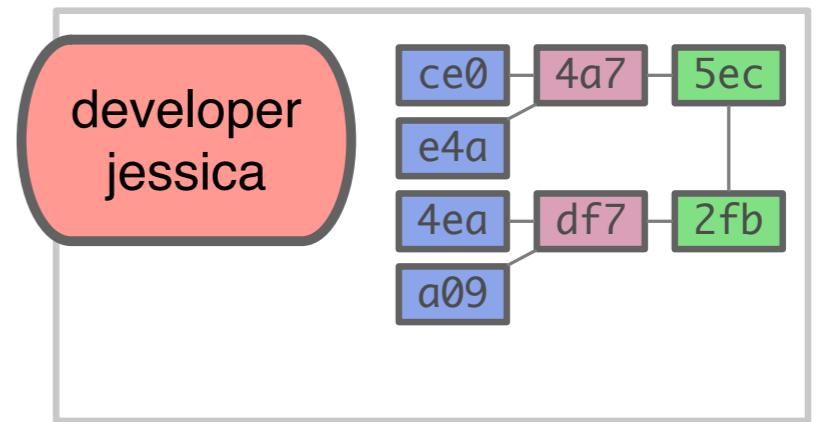
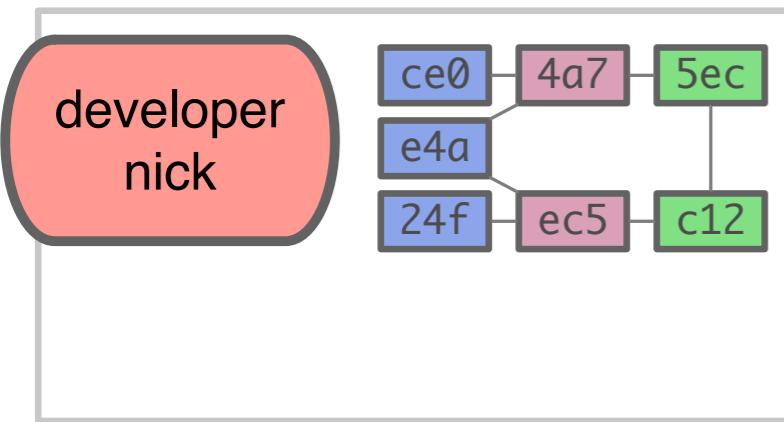


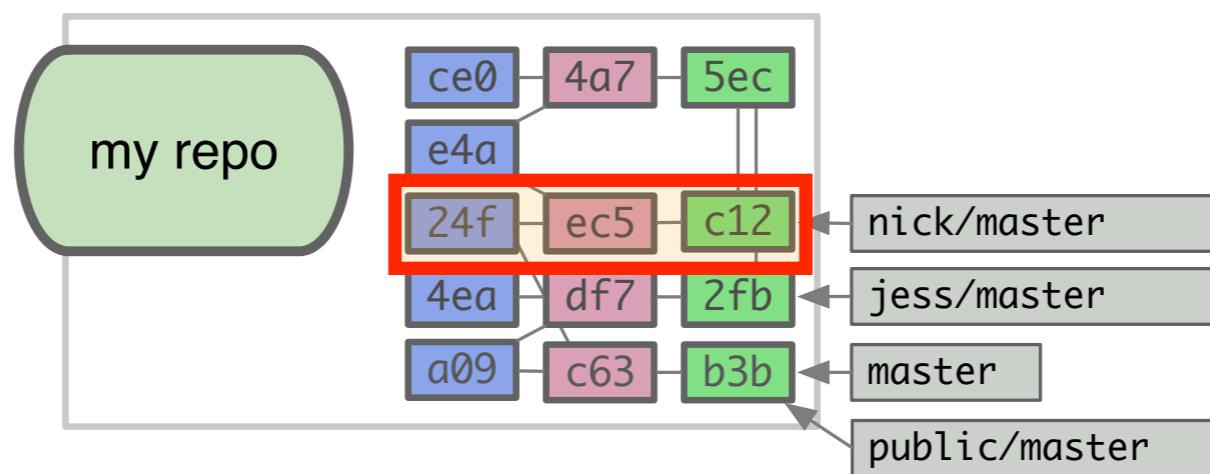
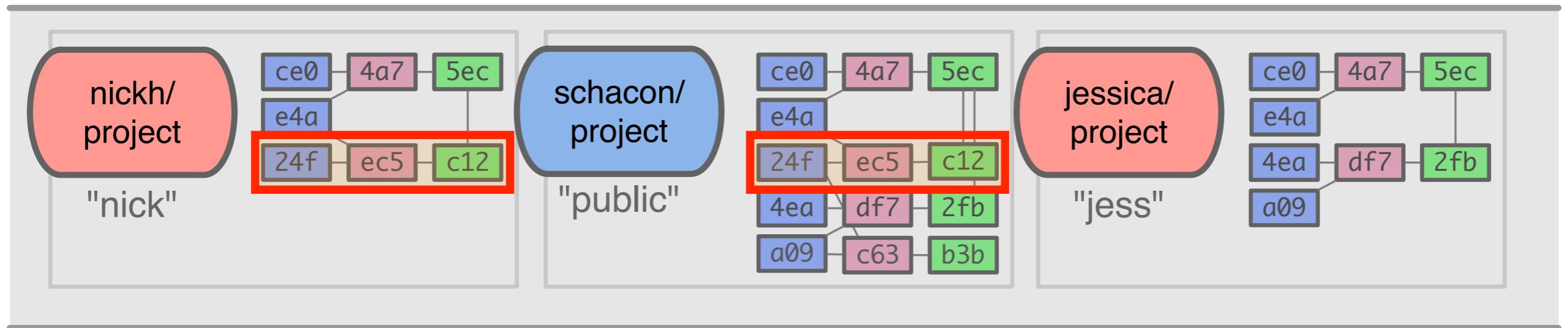
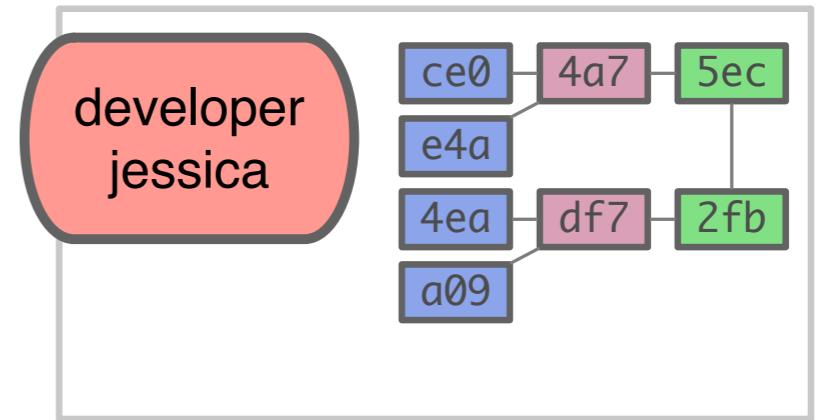
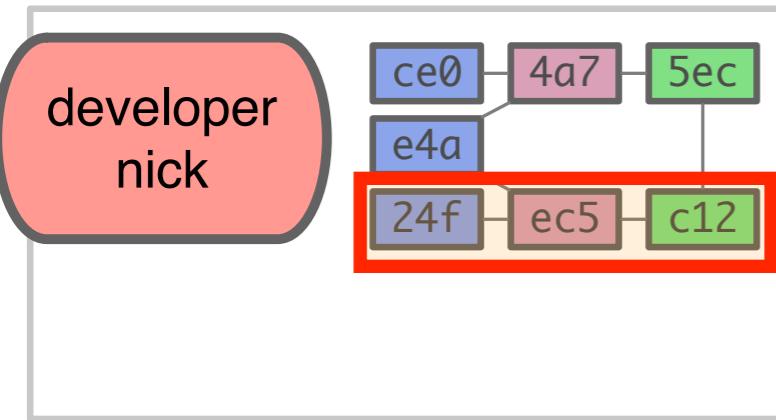


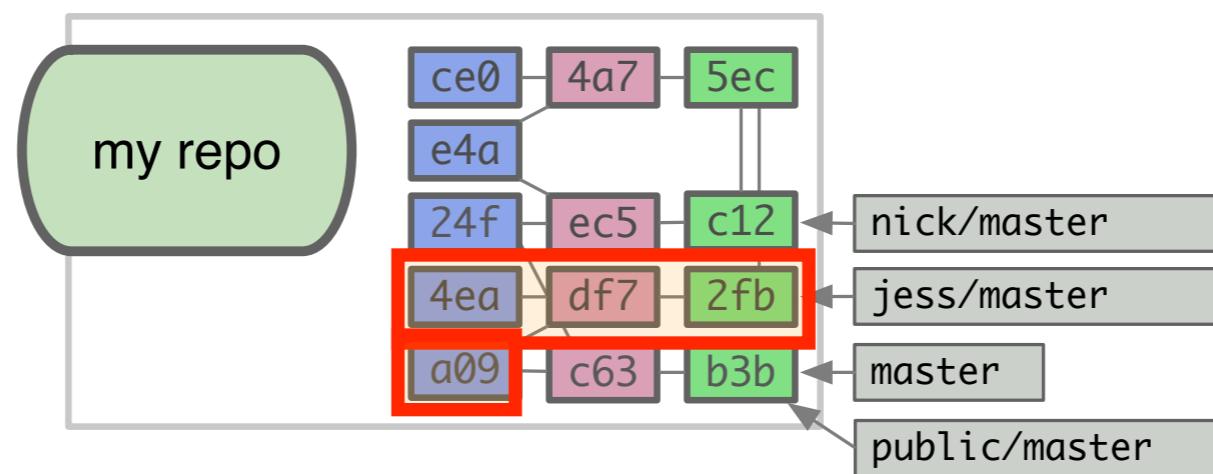
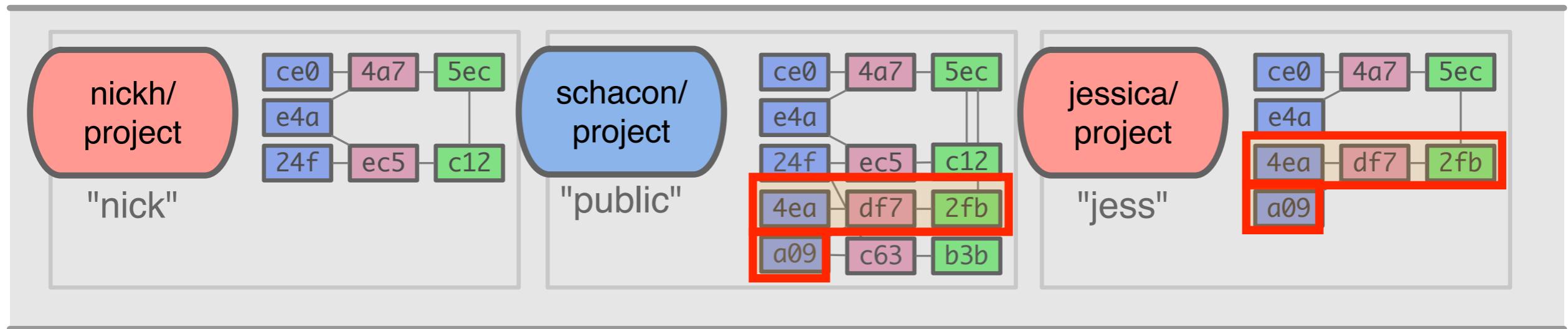
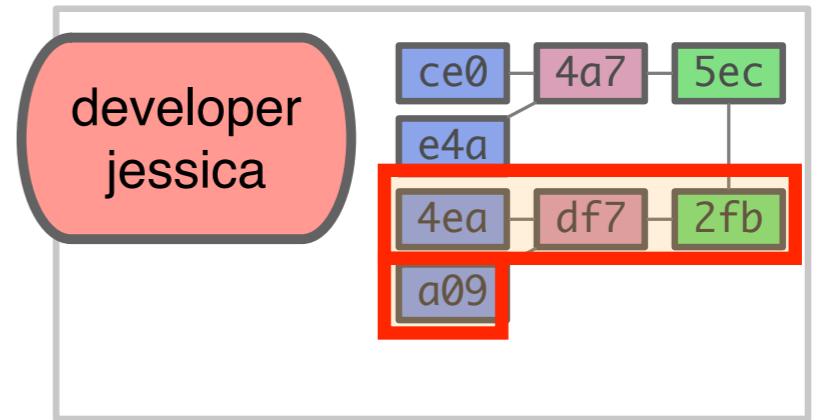
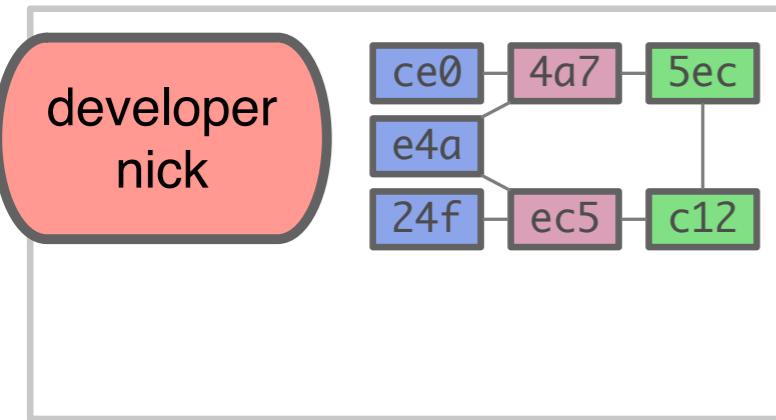
git merge nick/master jess/master

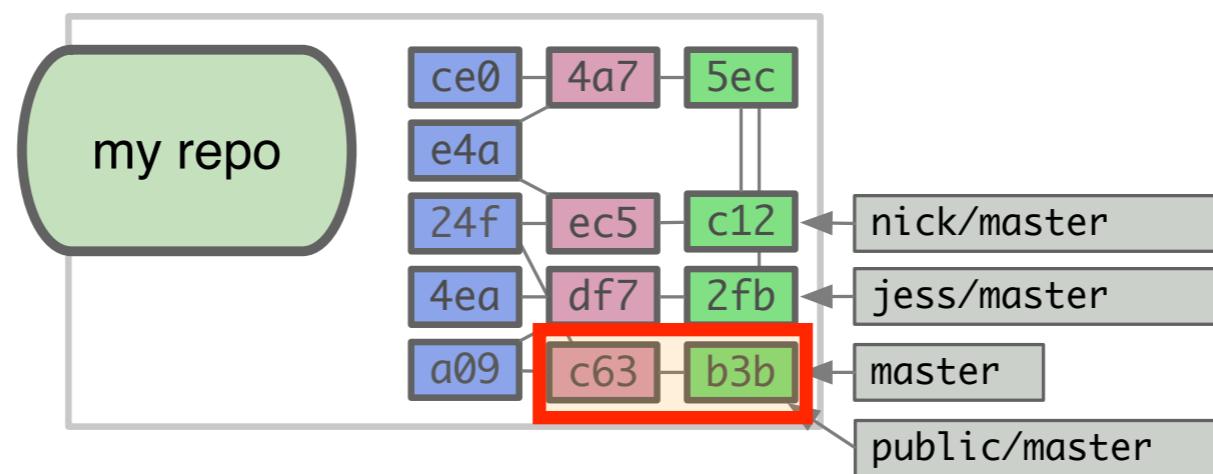
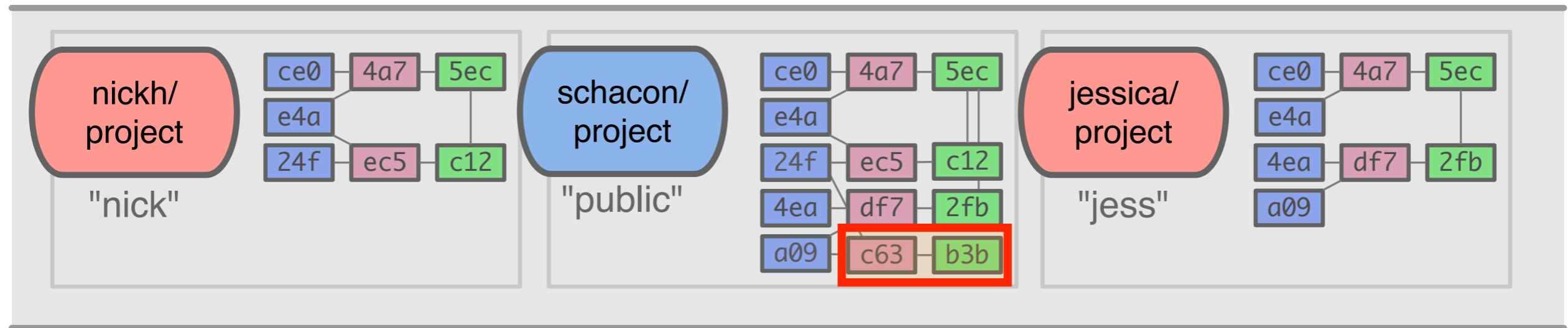
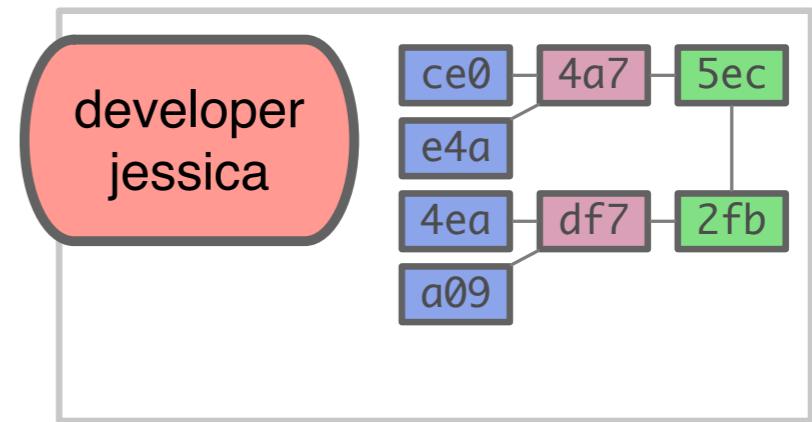
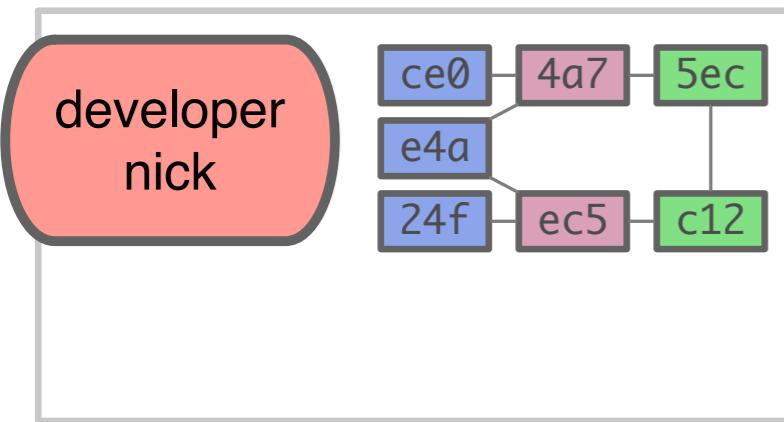


git push public





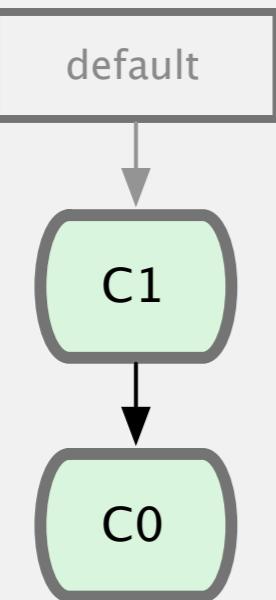




Remotes Are Branches

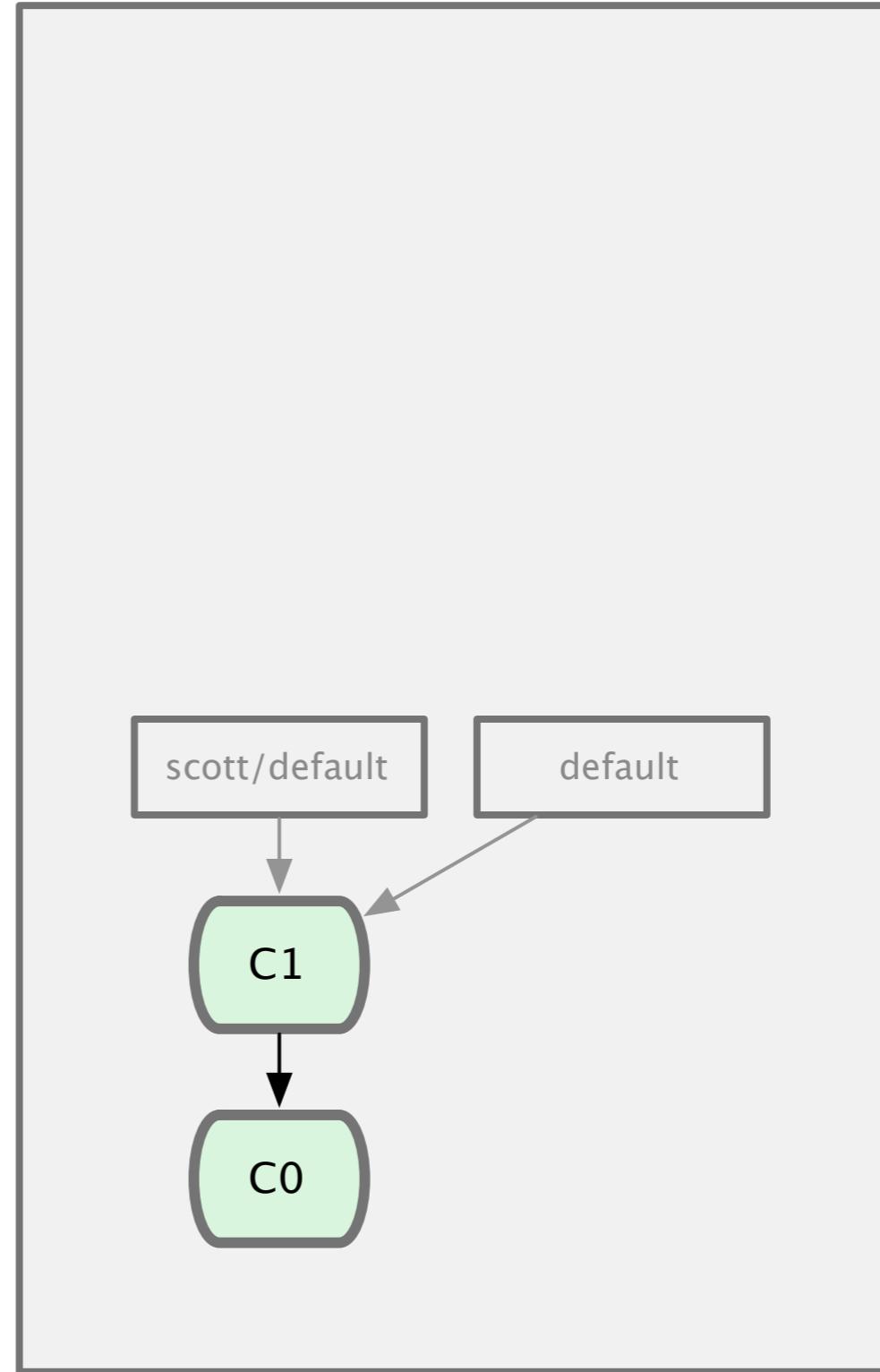
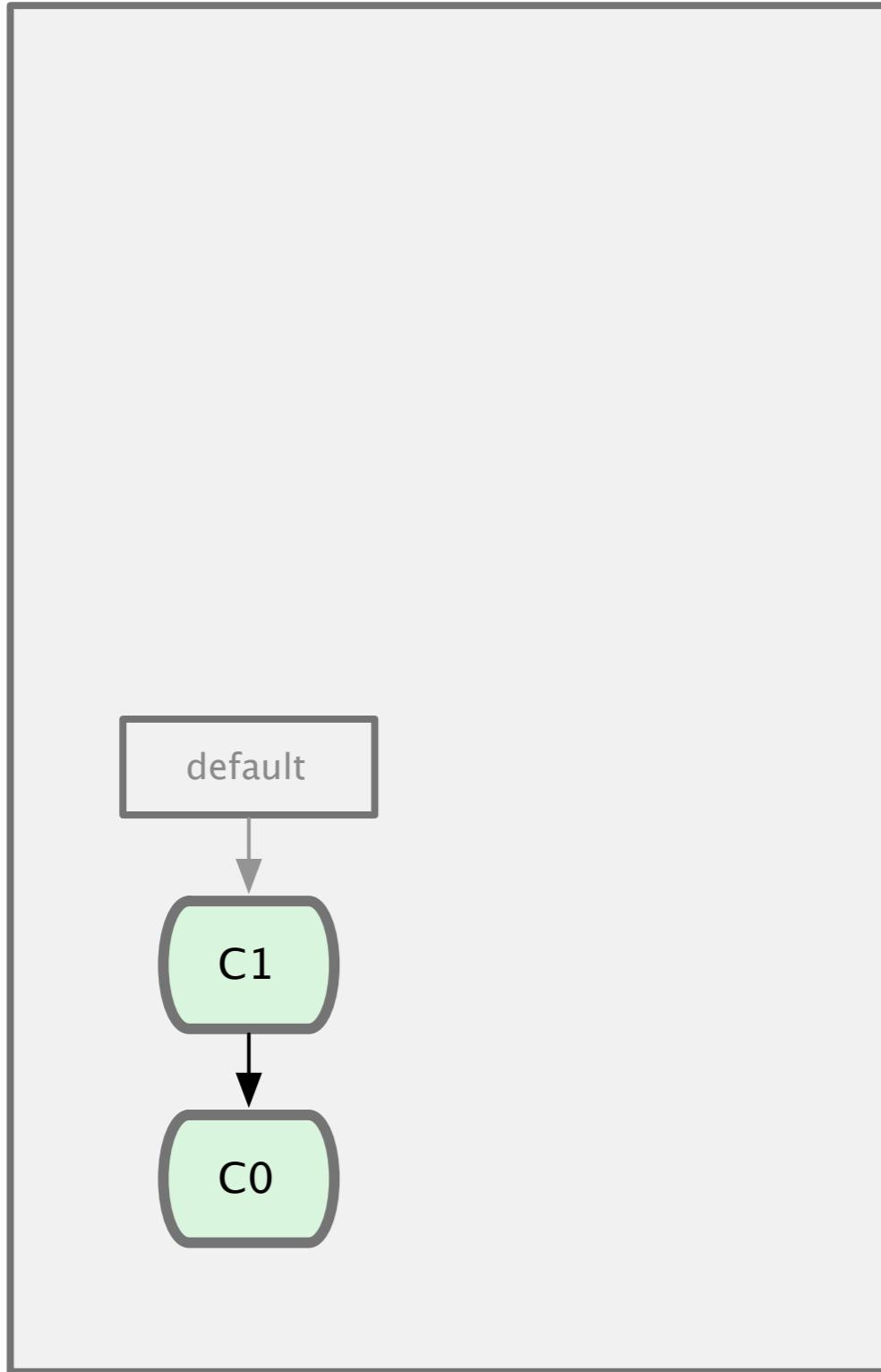
scott

jessica



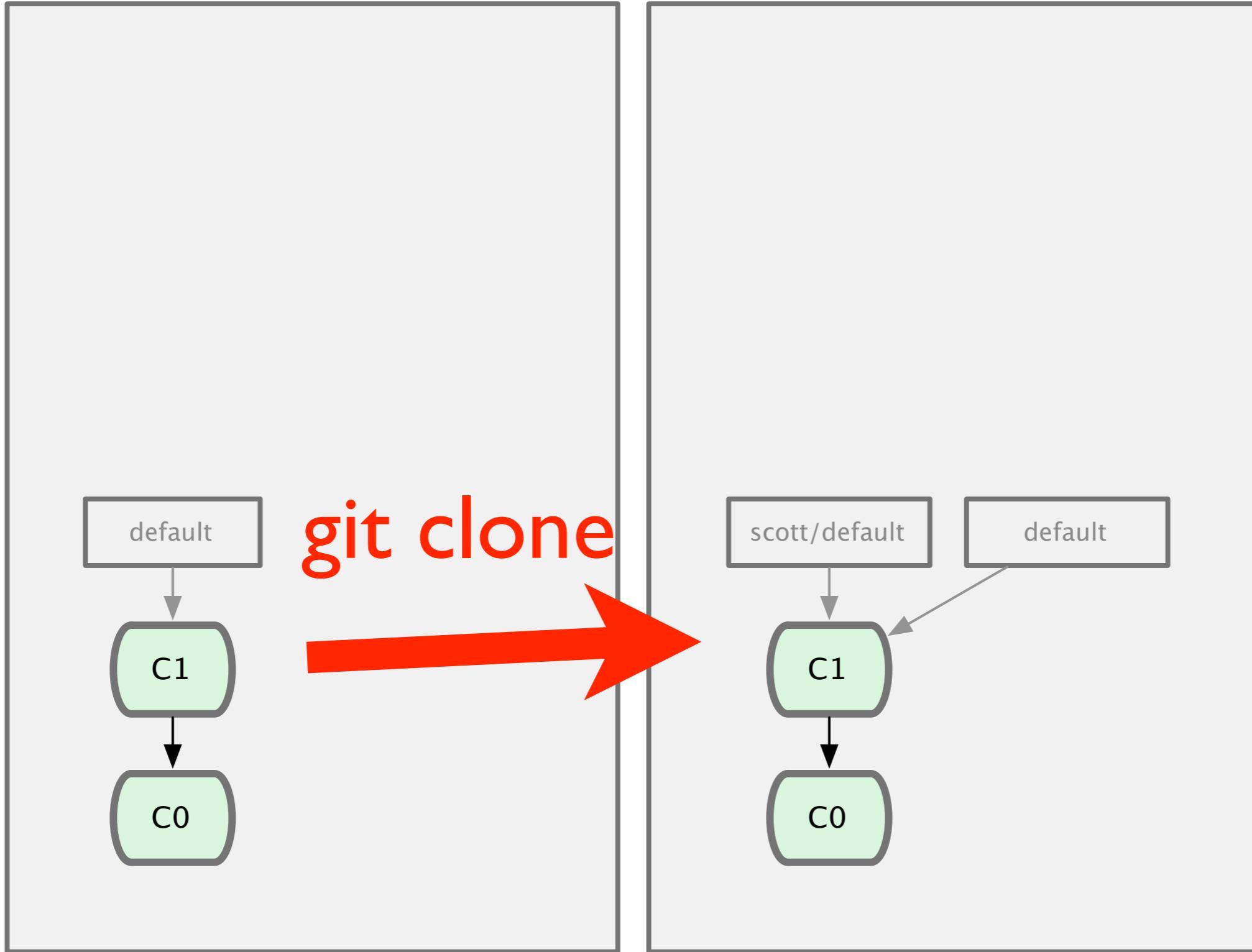
scott

jessica

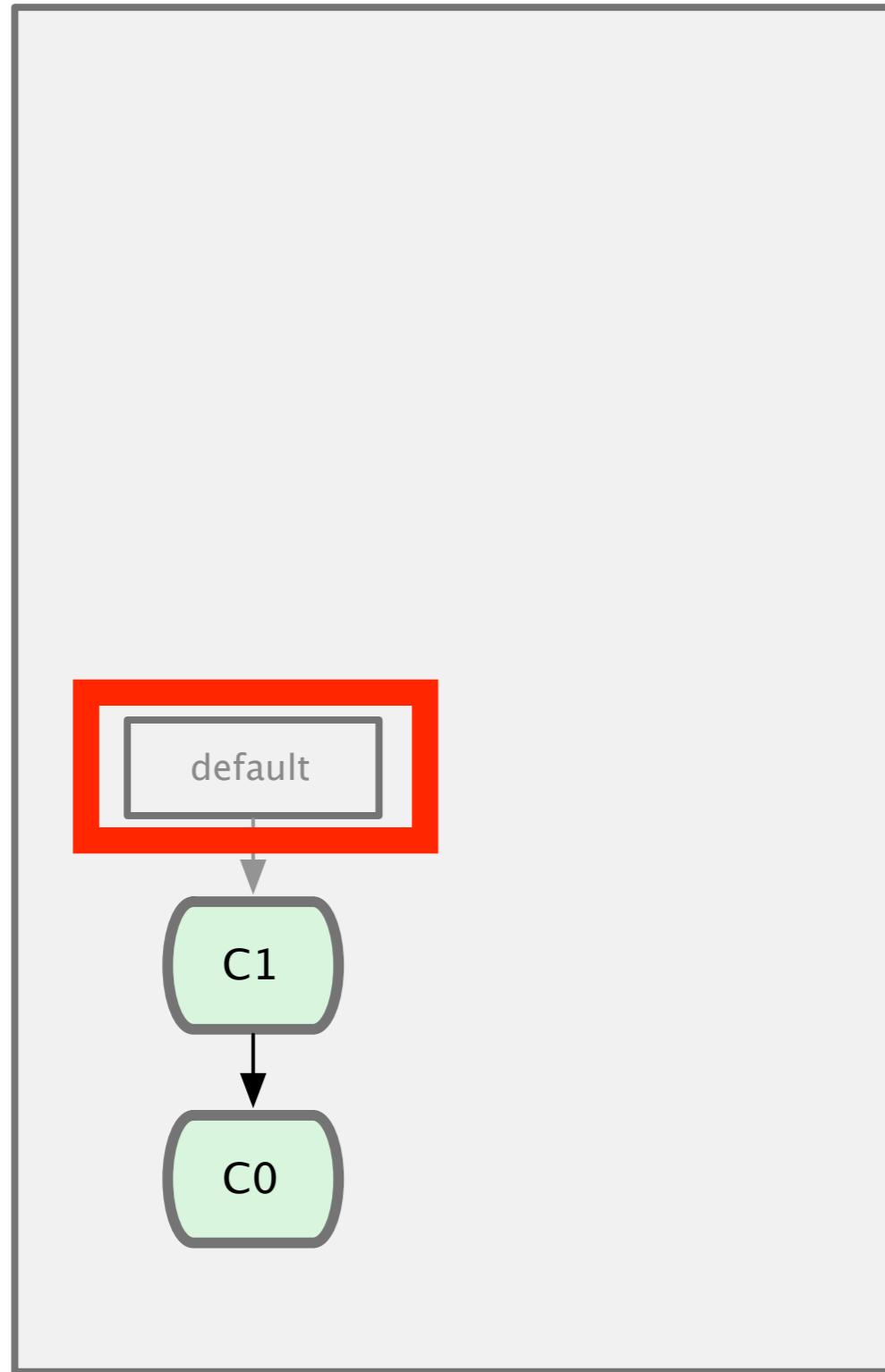


scott

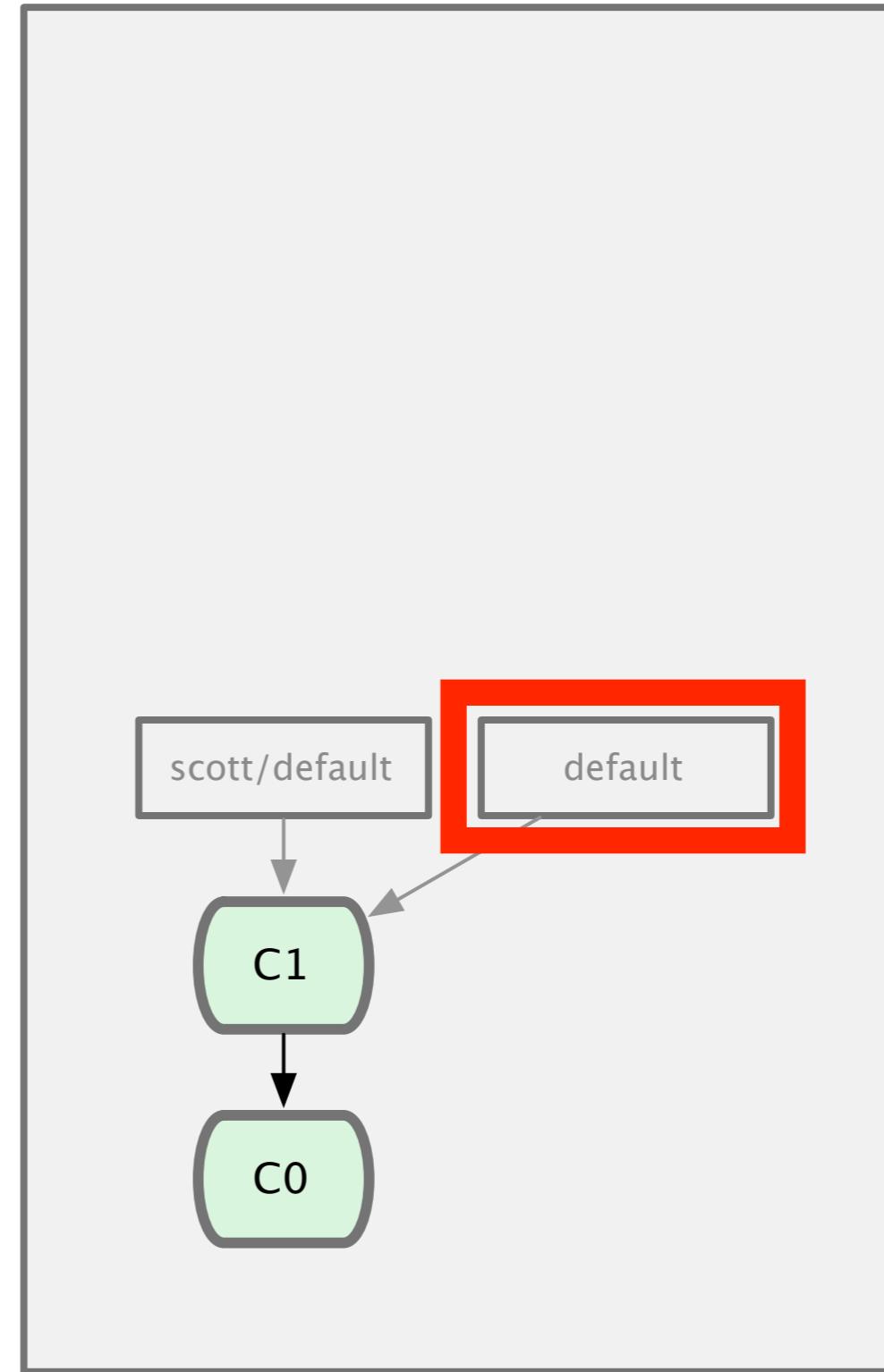
jessica



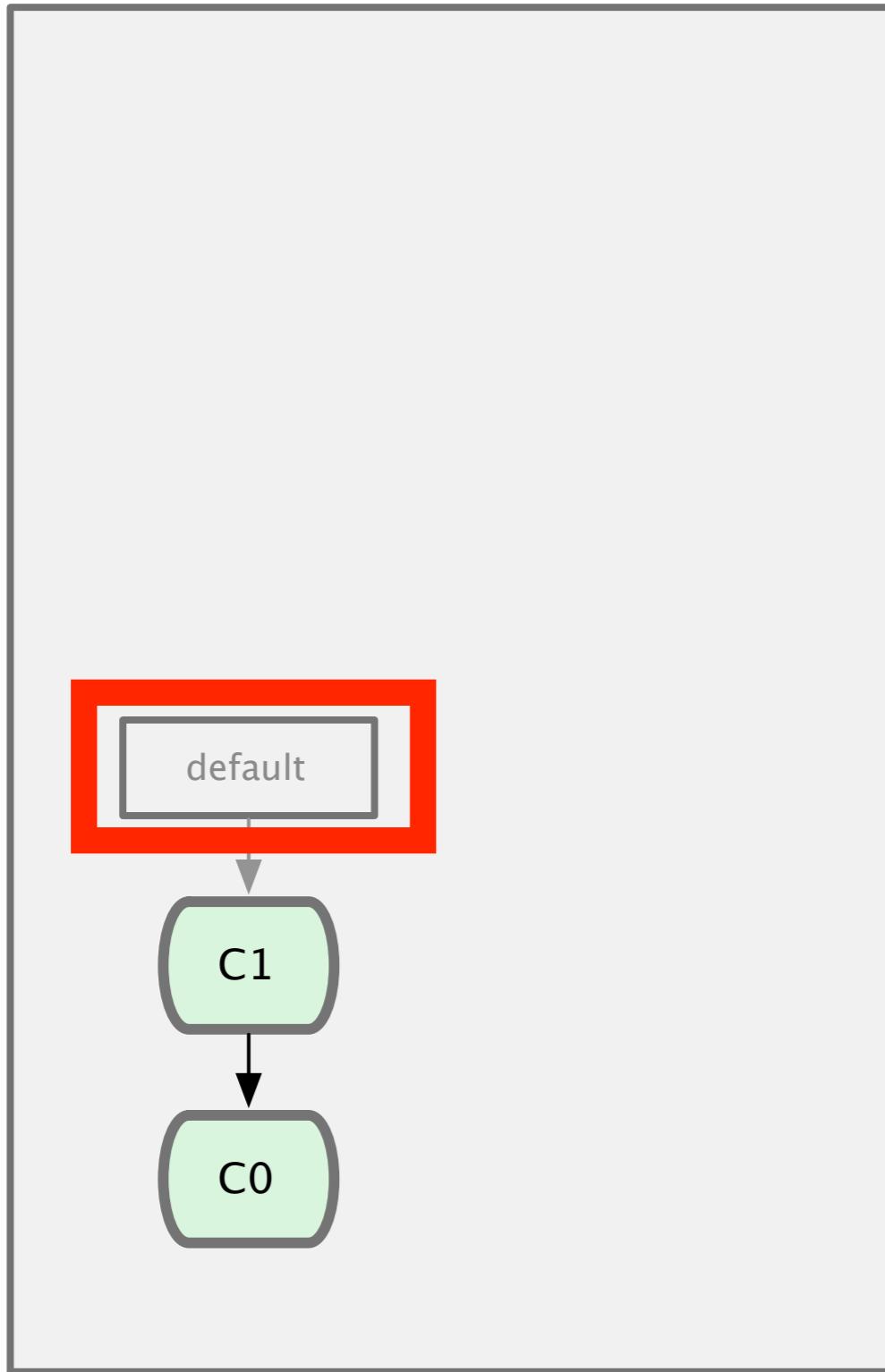
scott



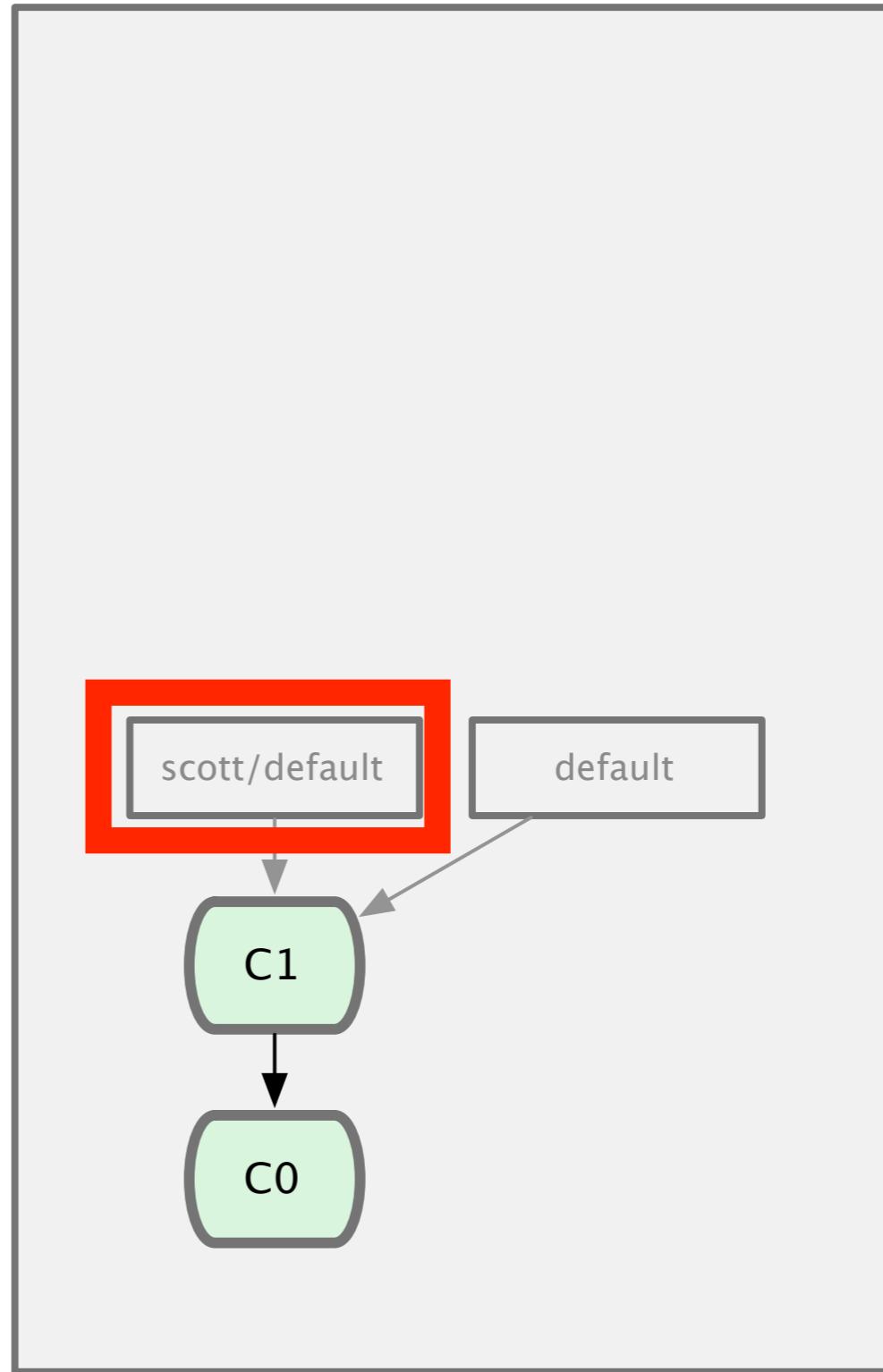
jessica



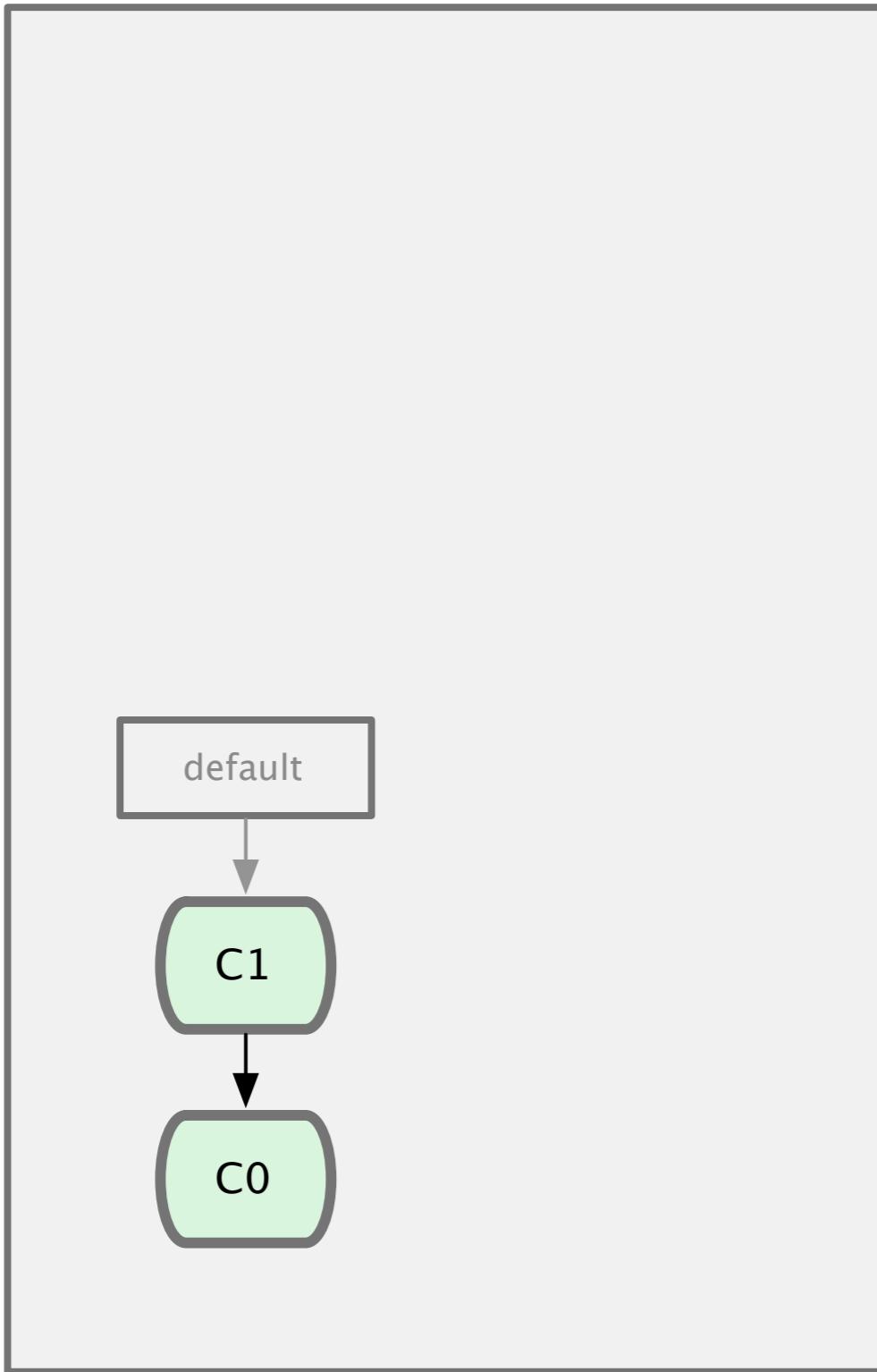
scott



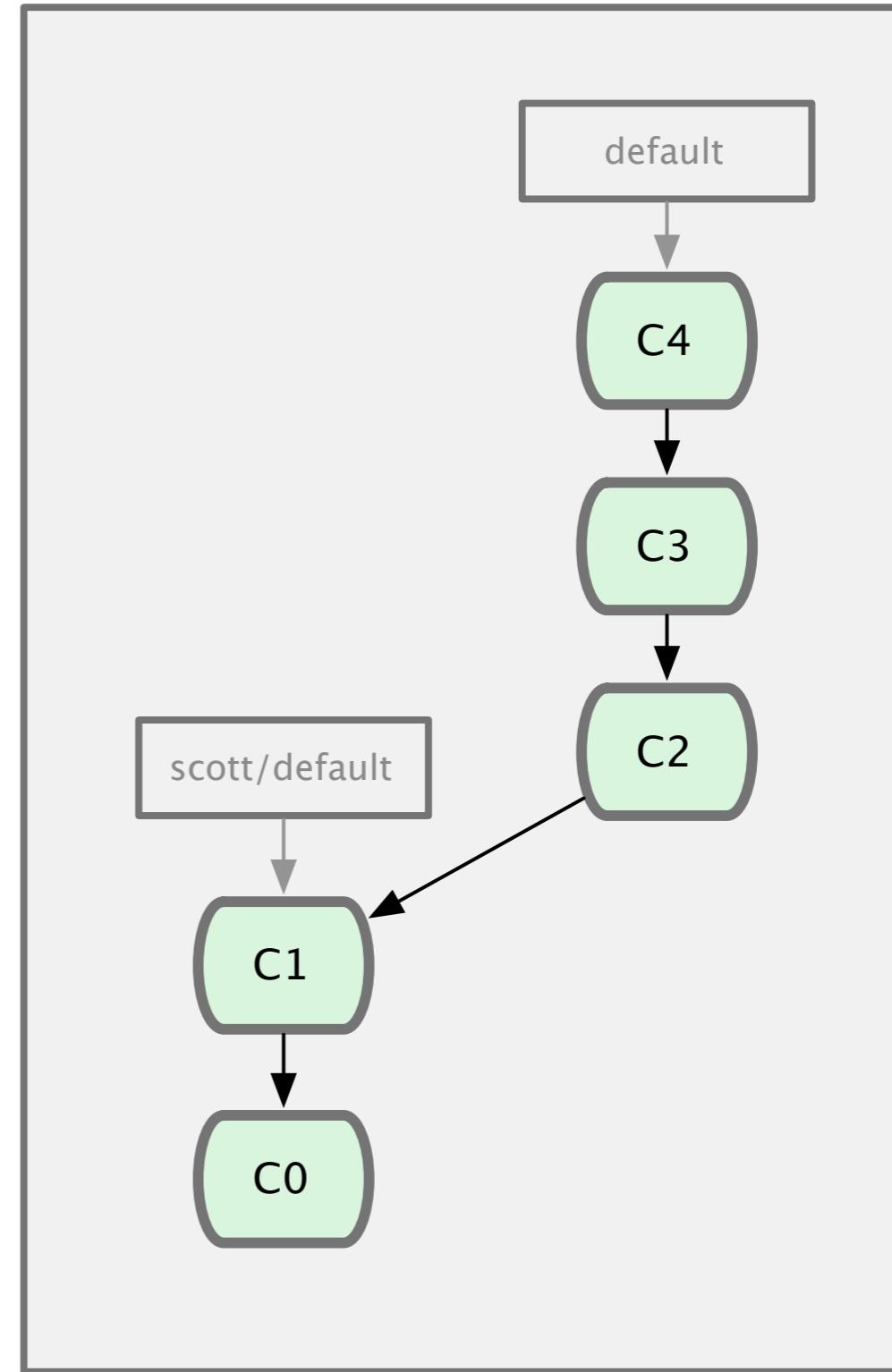
jessica



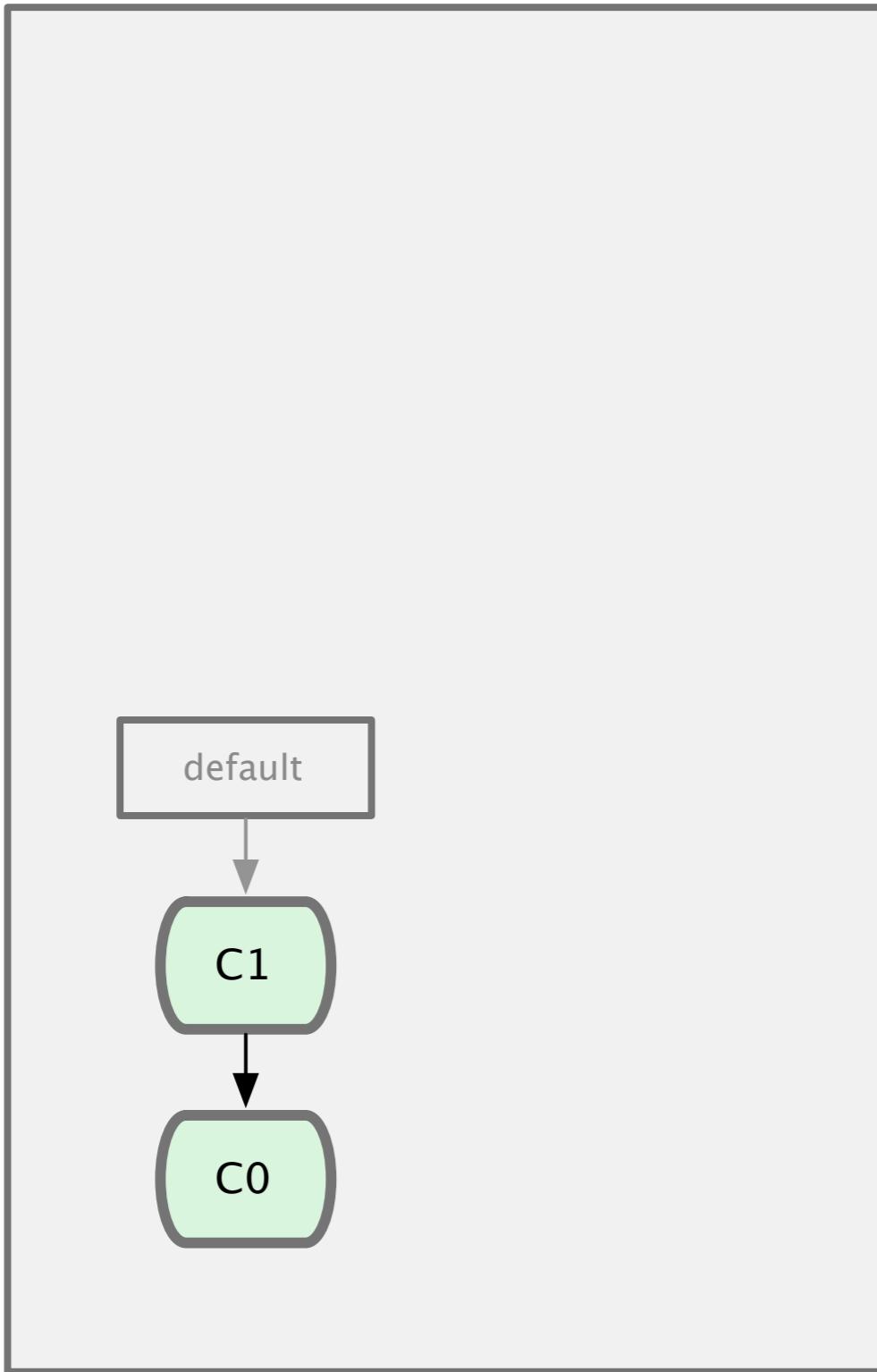
scott



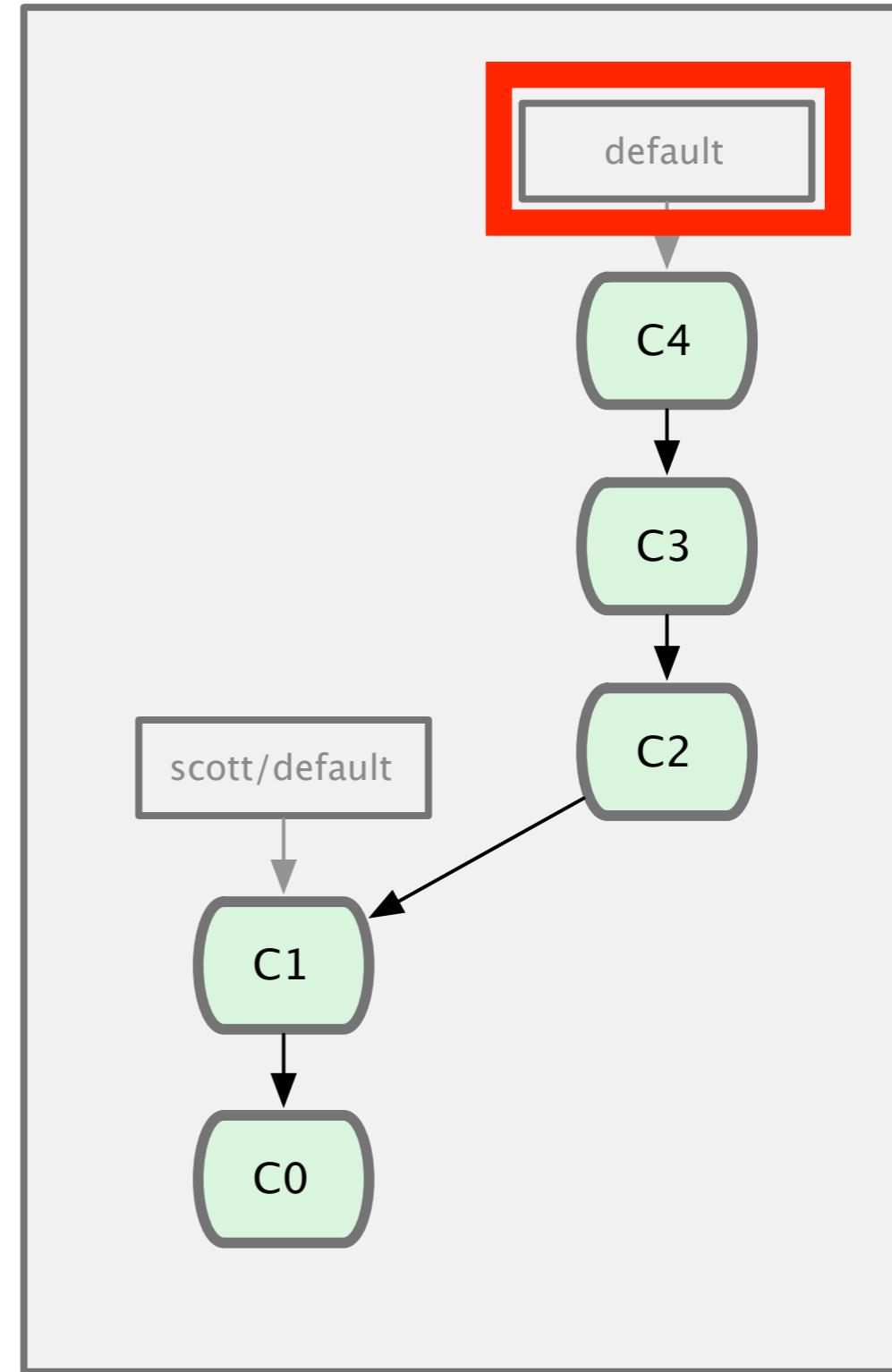
jessica



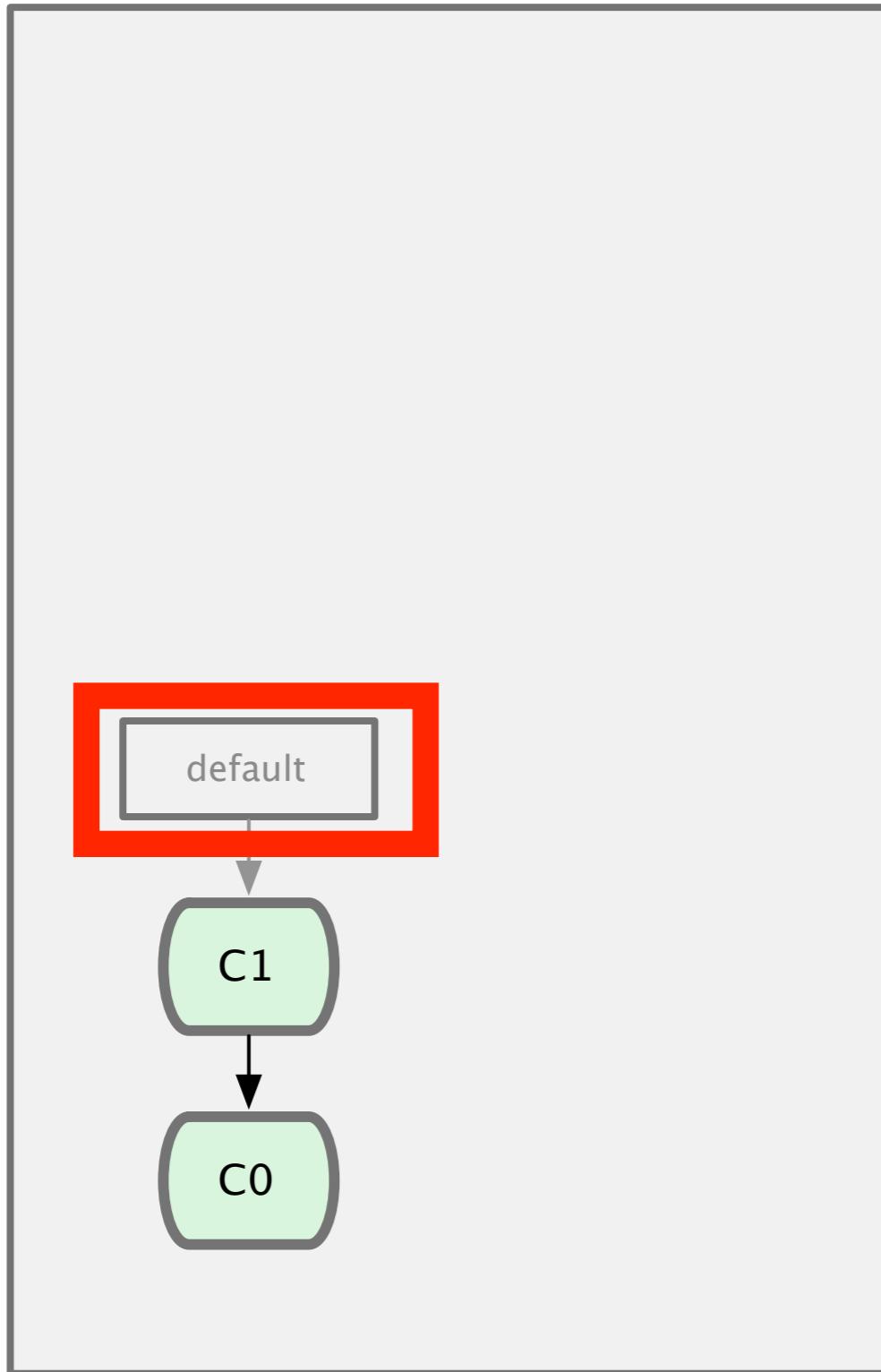
scott



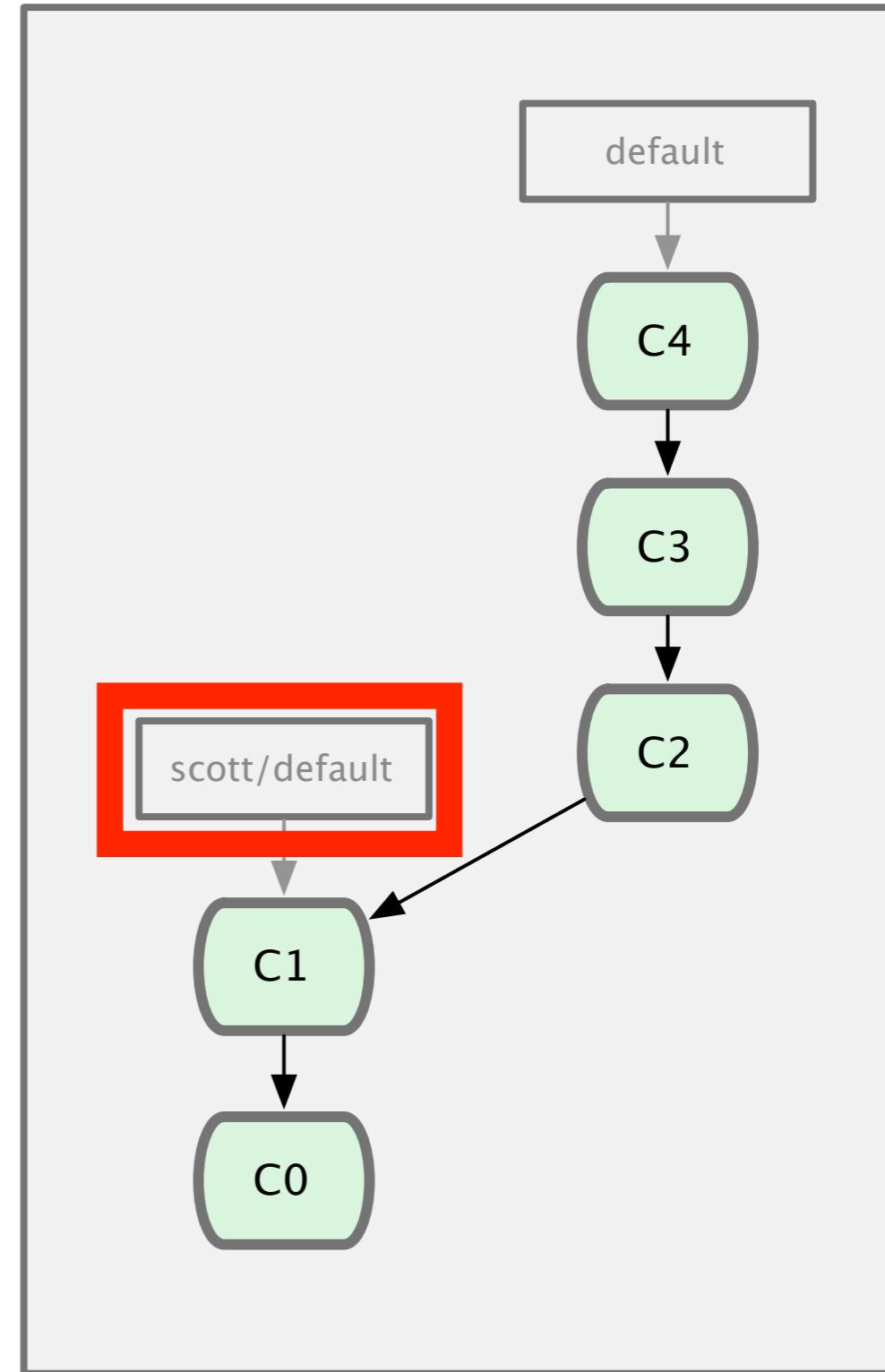
jessica



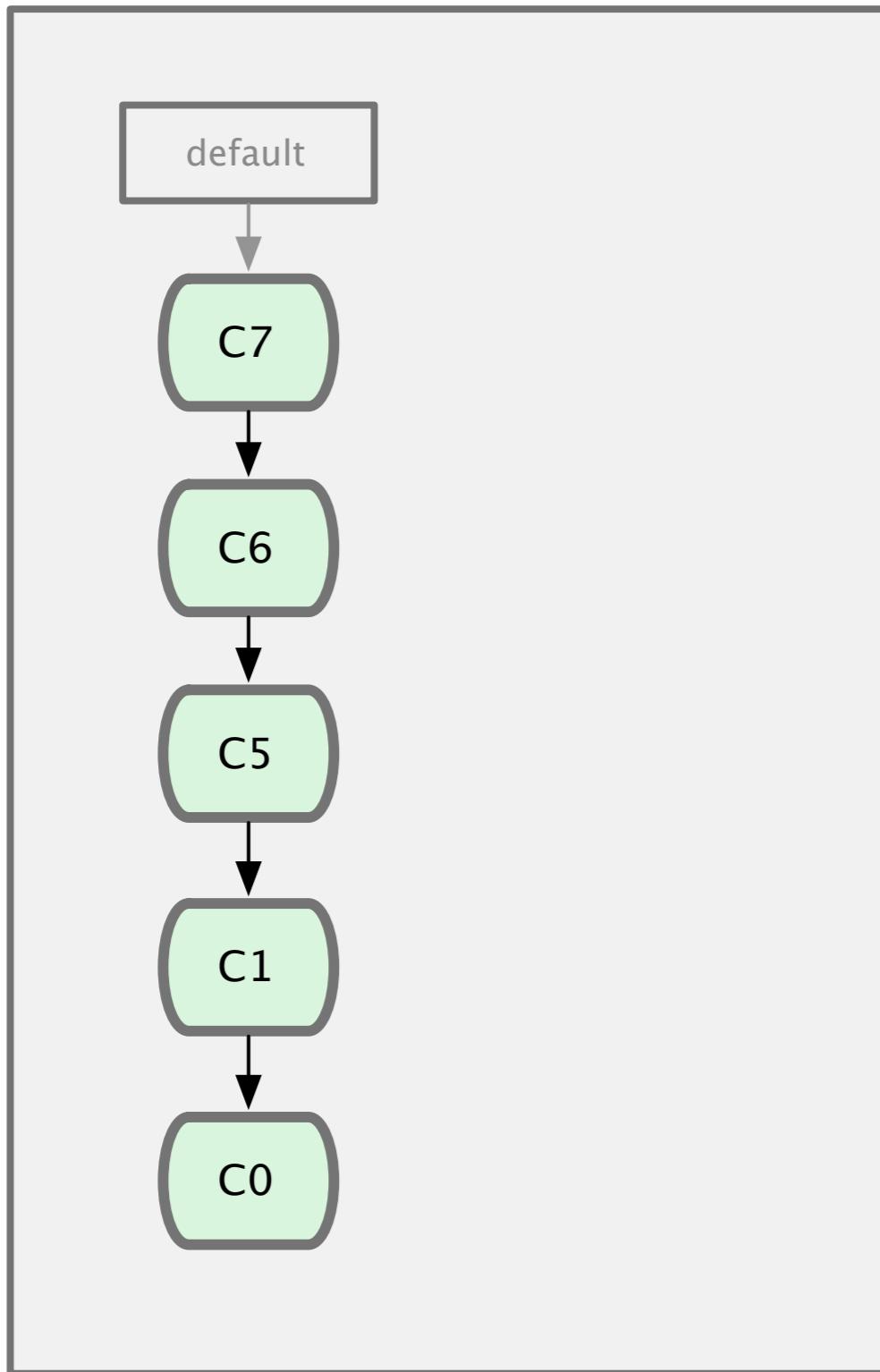
scott



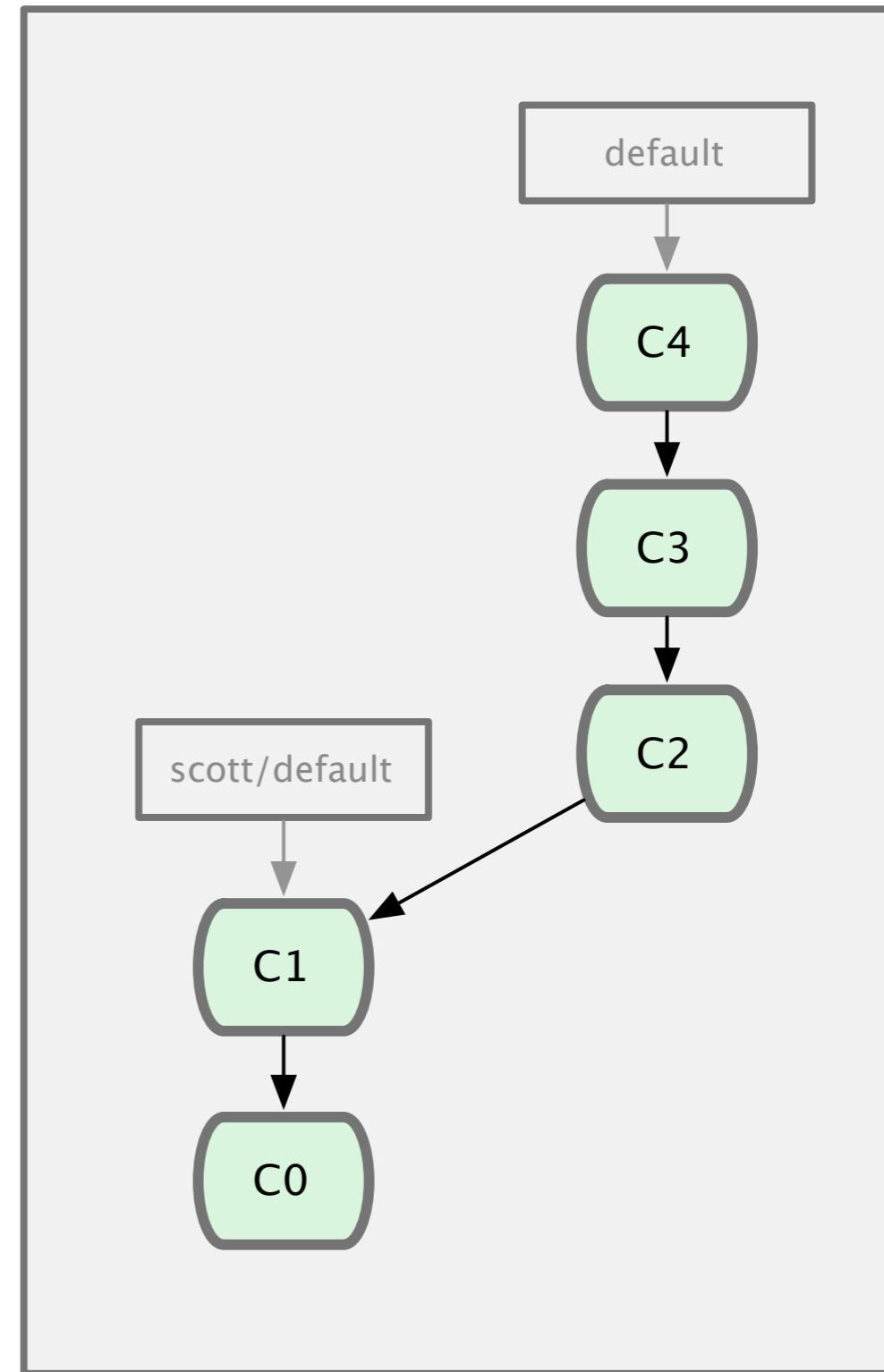
jessica



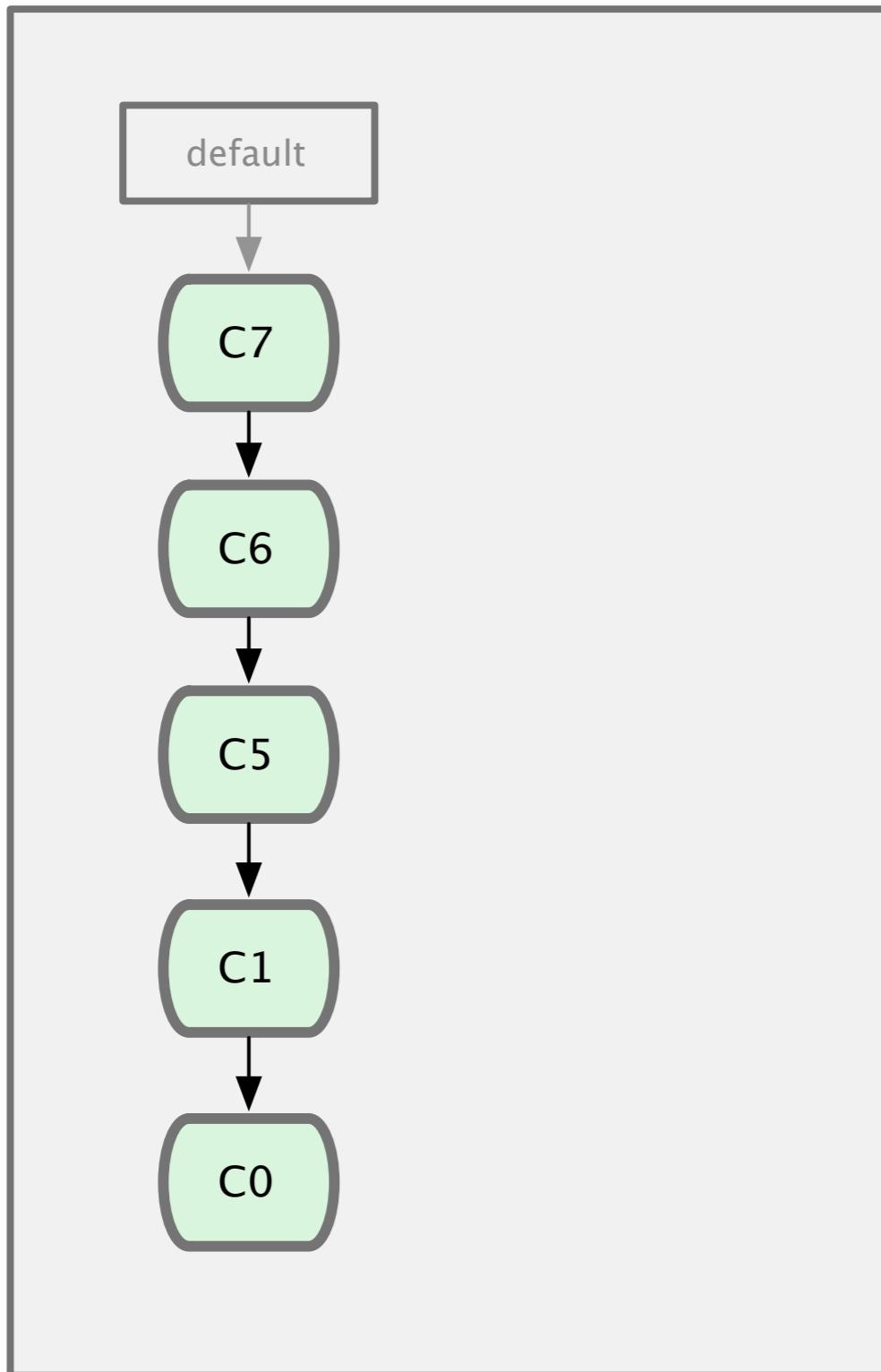
scott



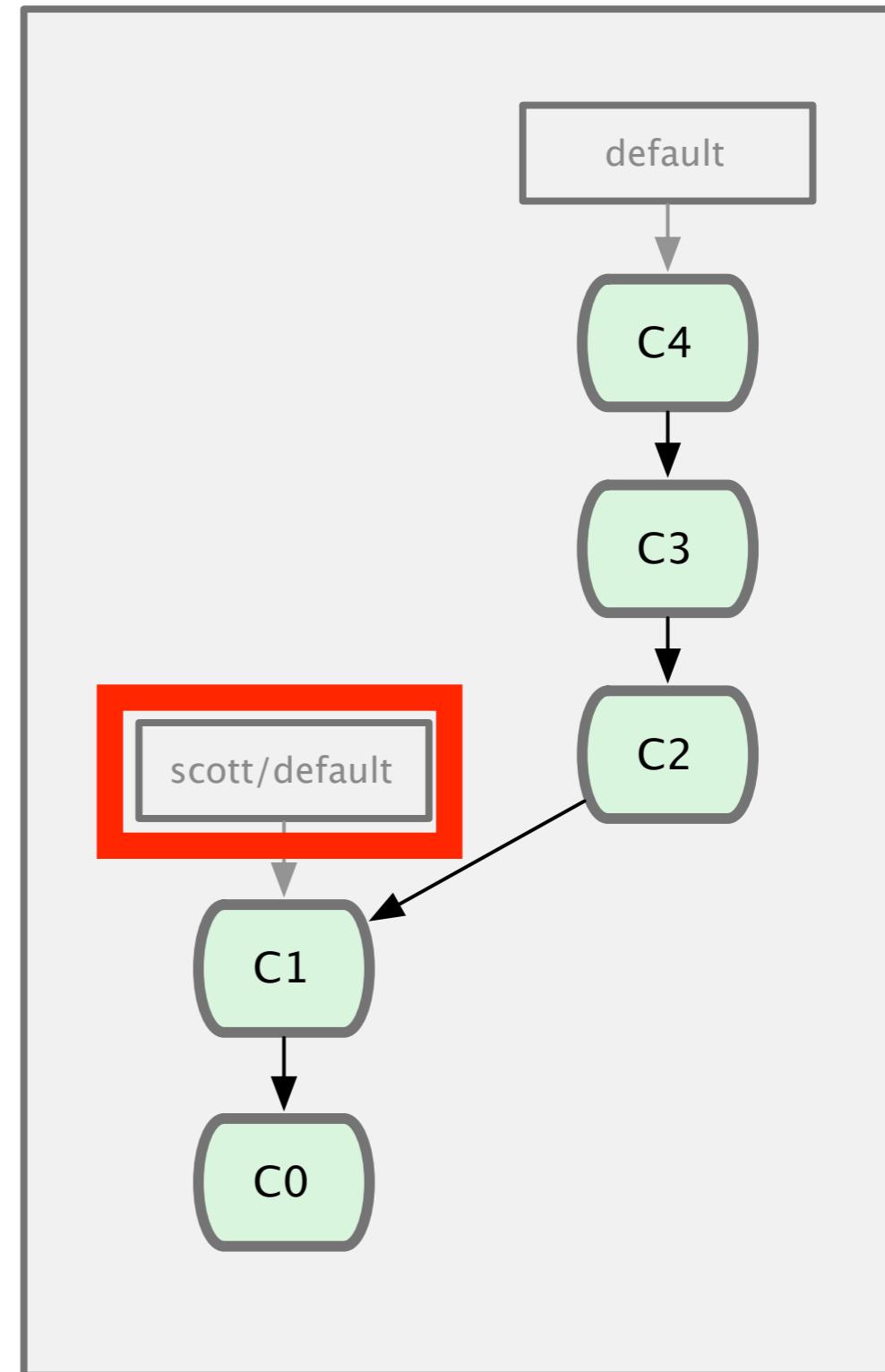
jessica



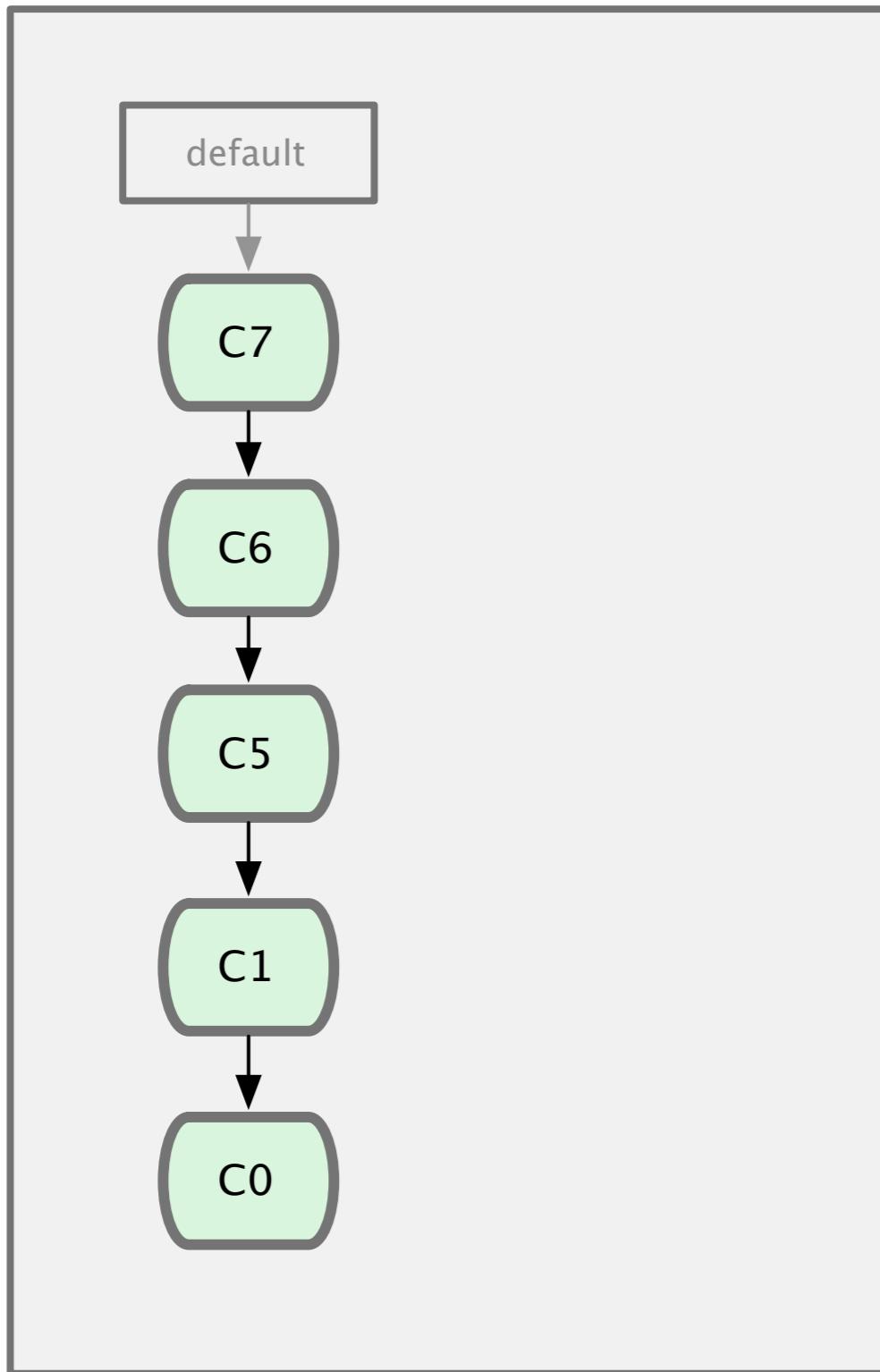
scott



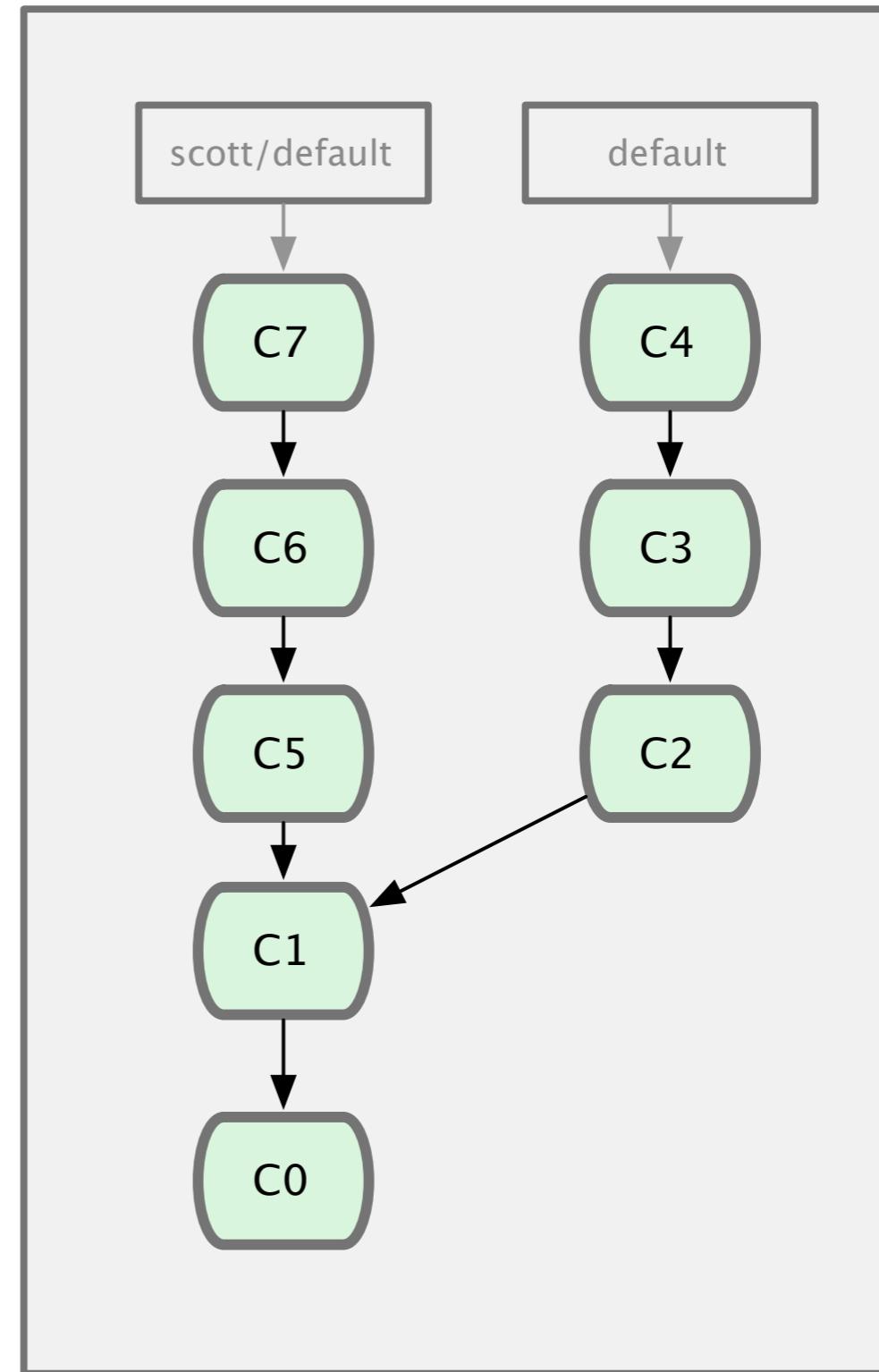
jessica



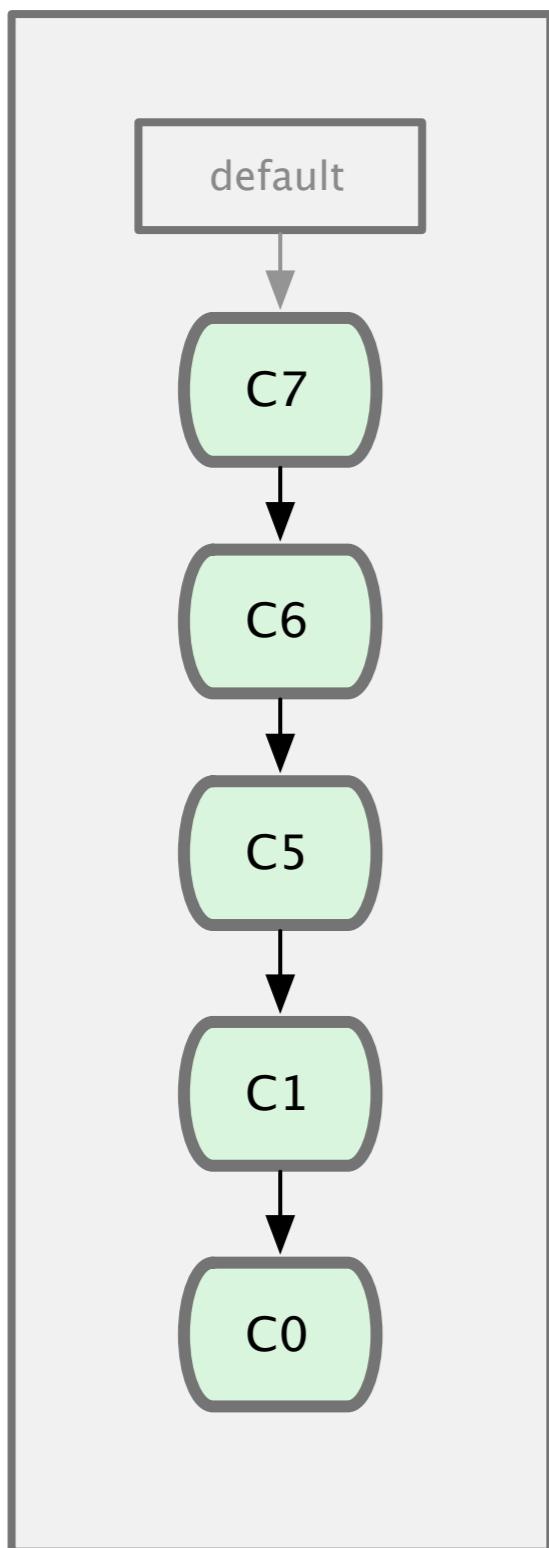
scott



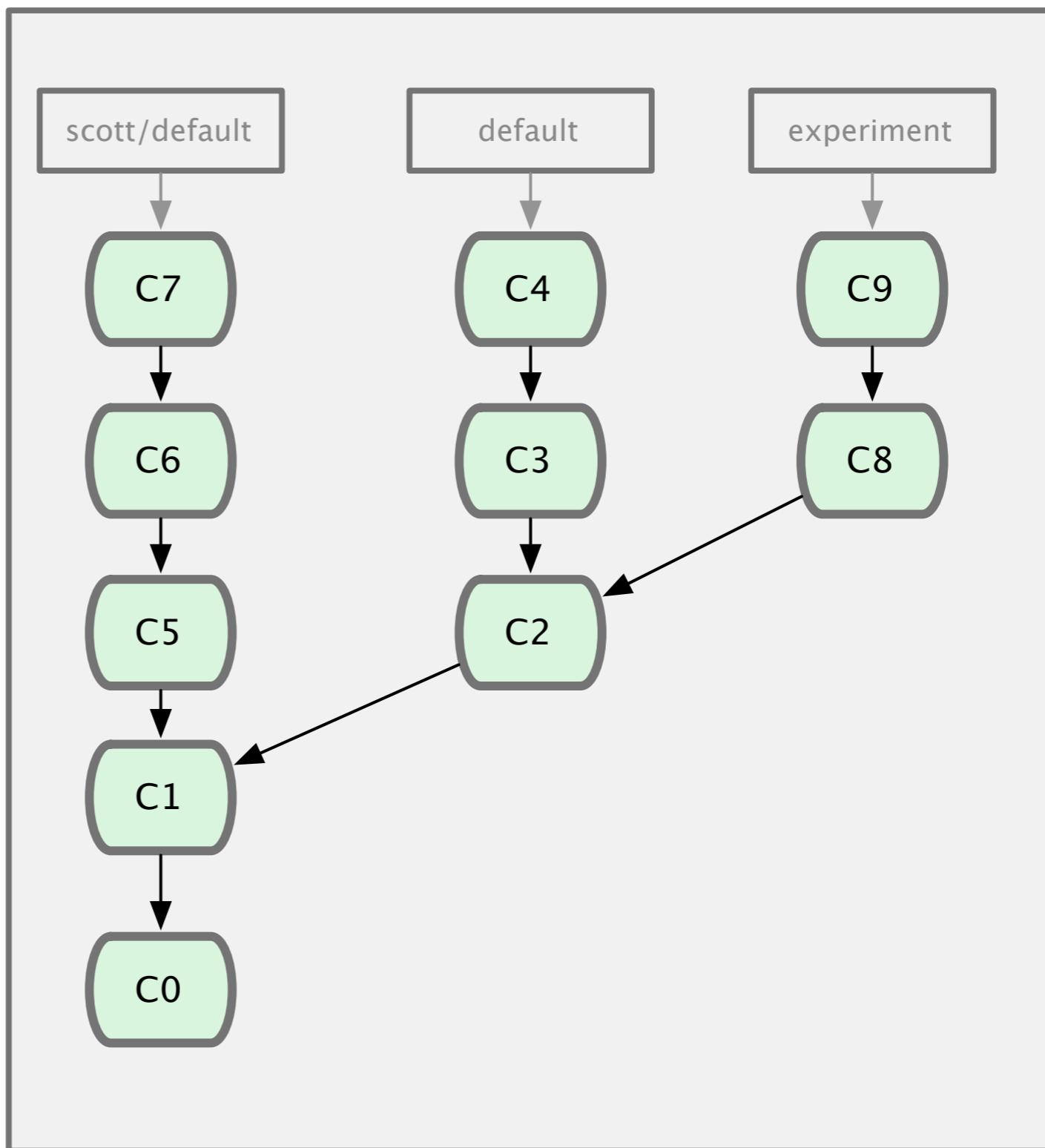
jessica



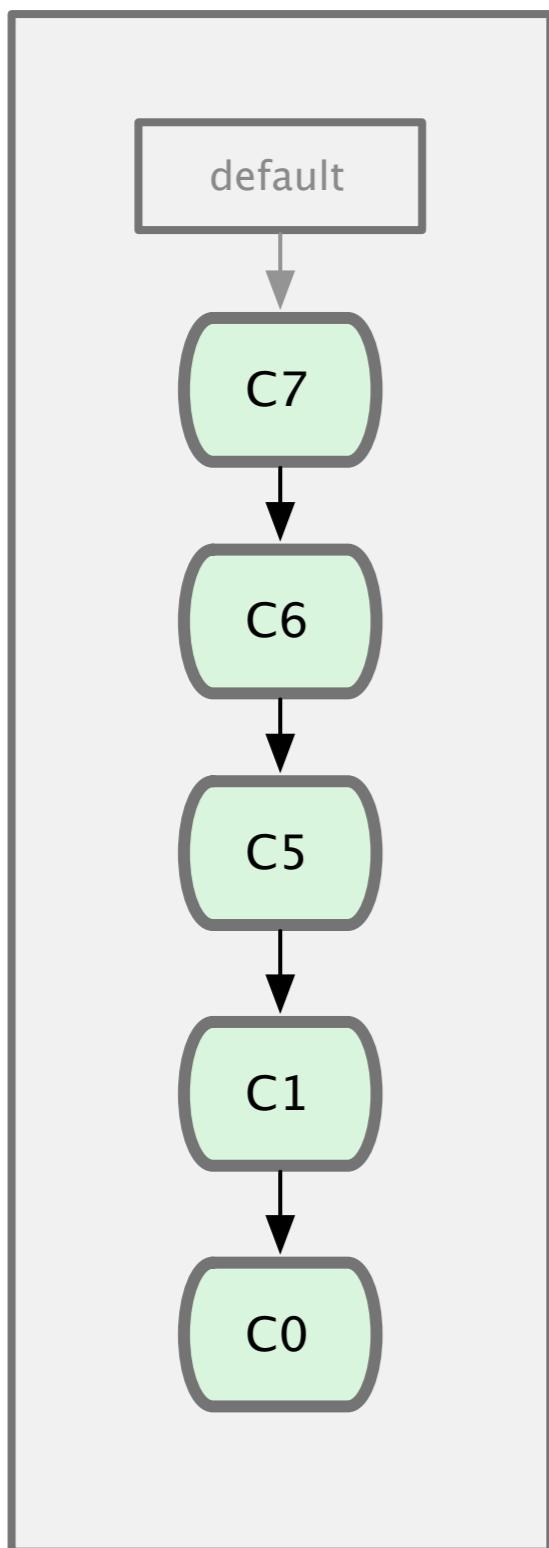
scott



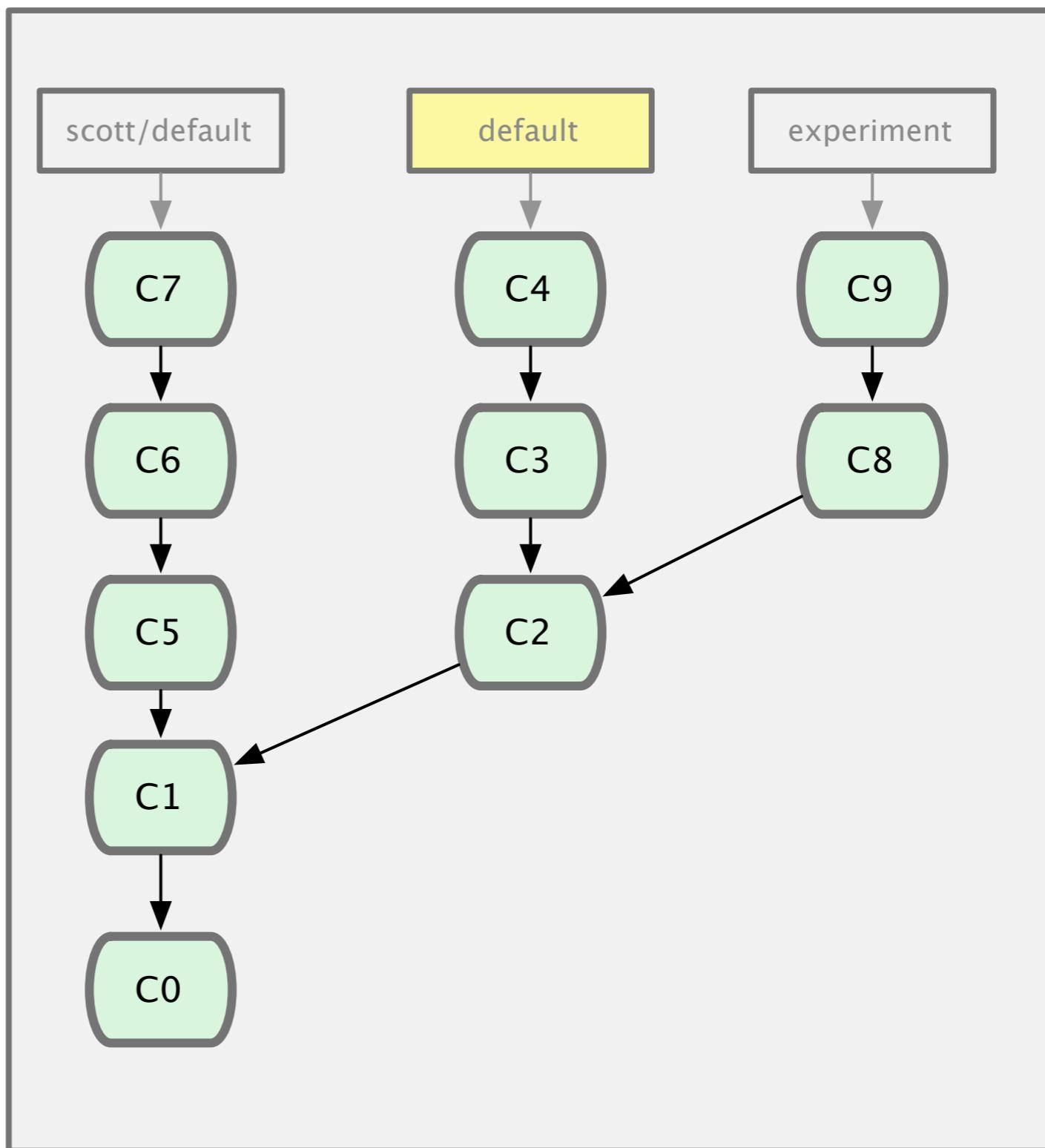
jessica



scott

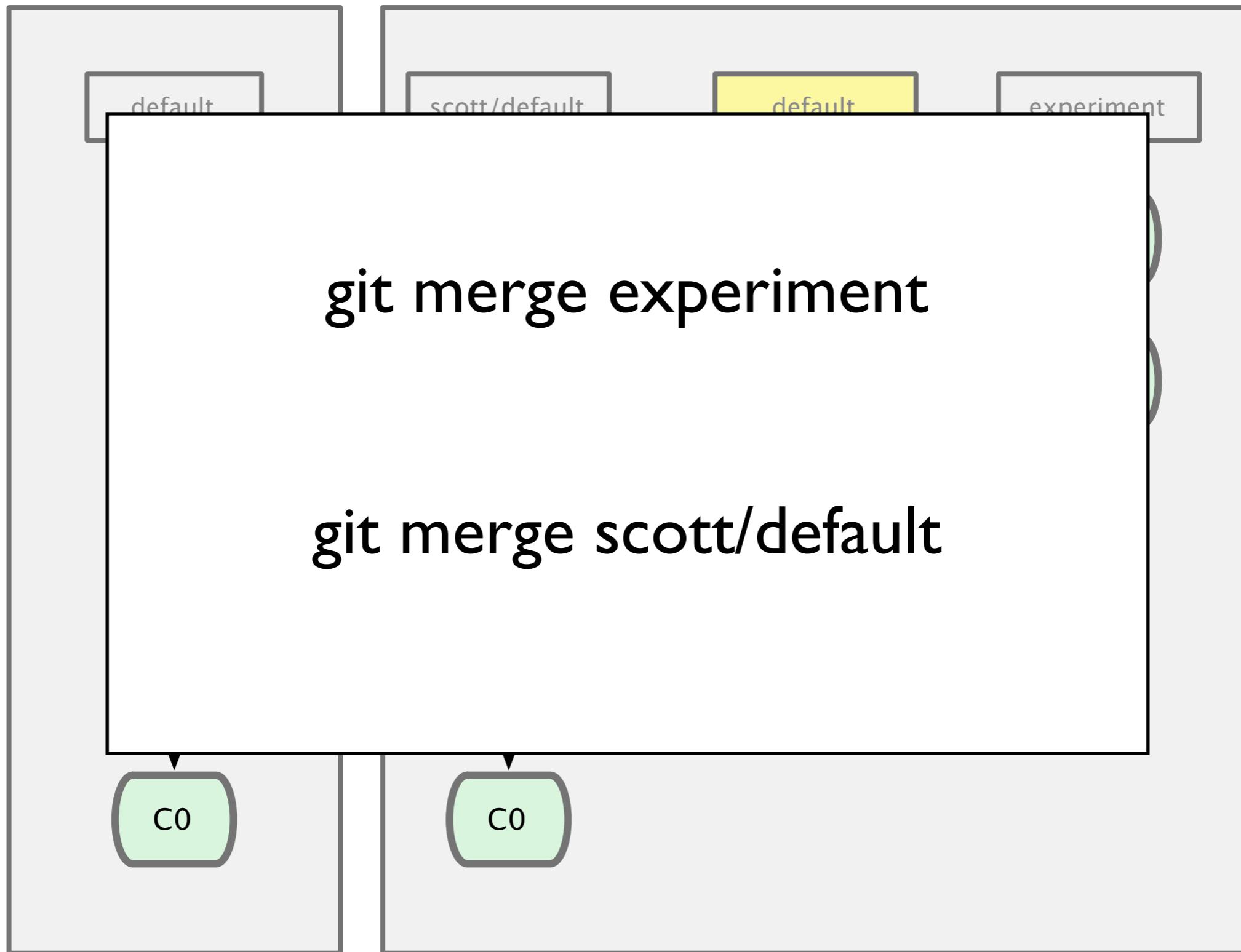


jessica

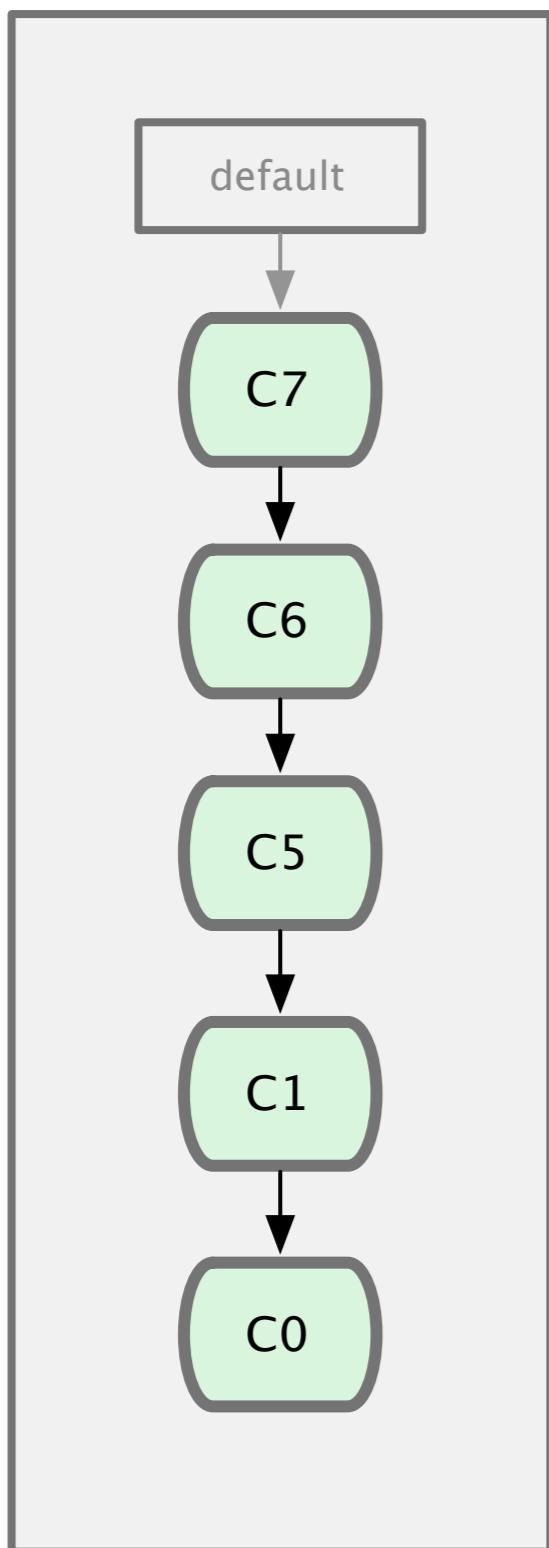


scott

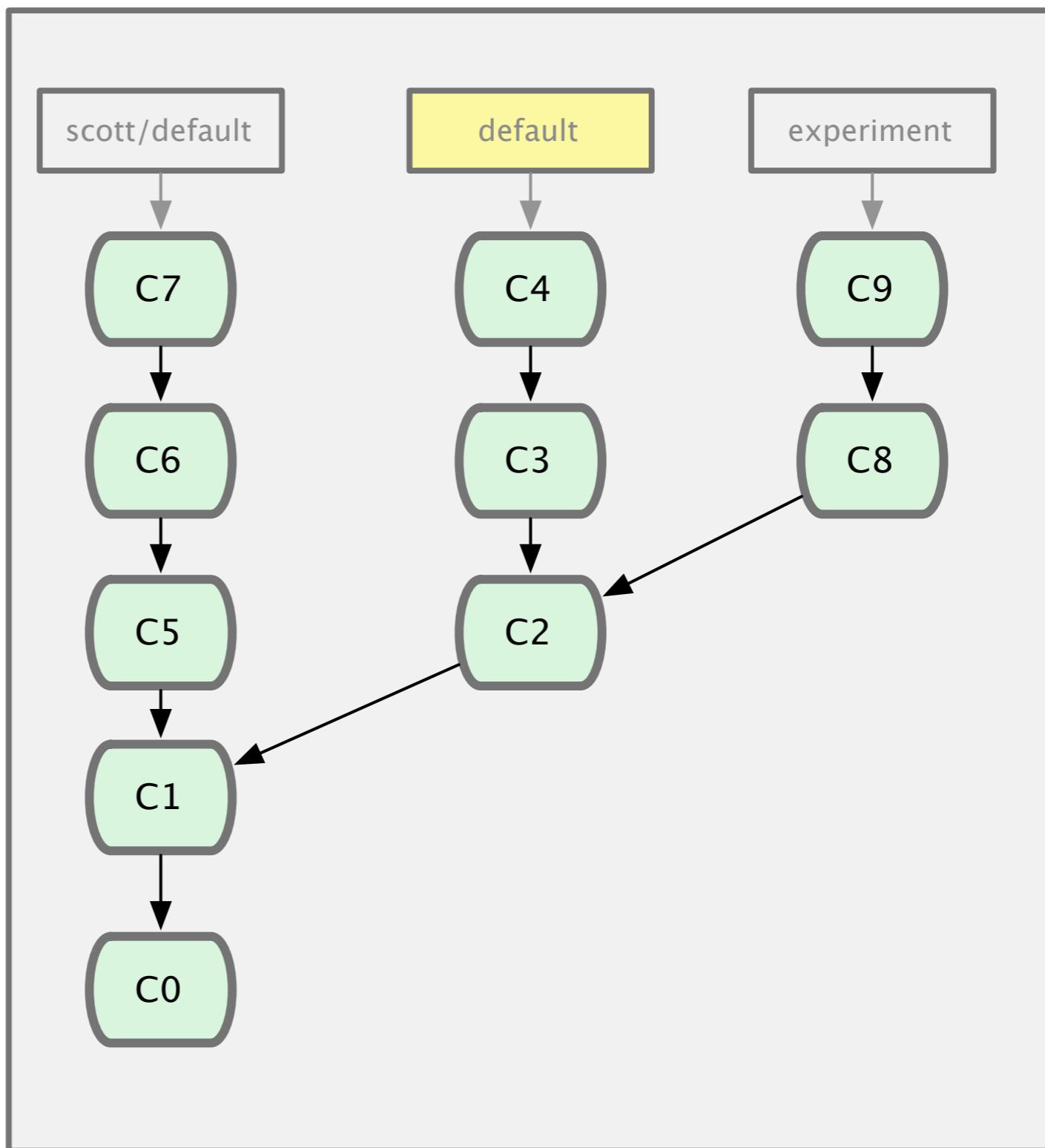
jessica



scott

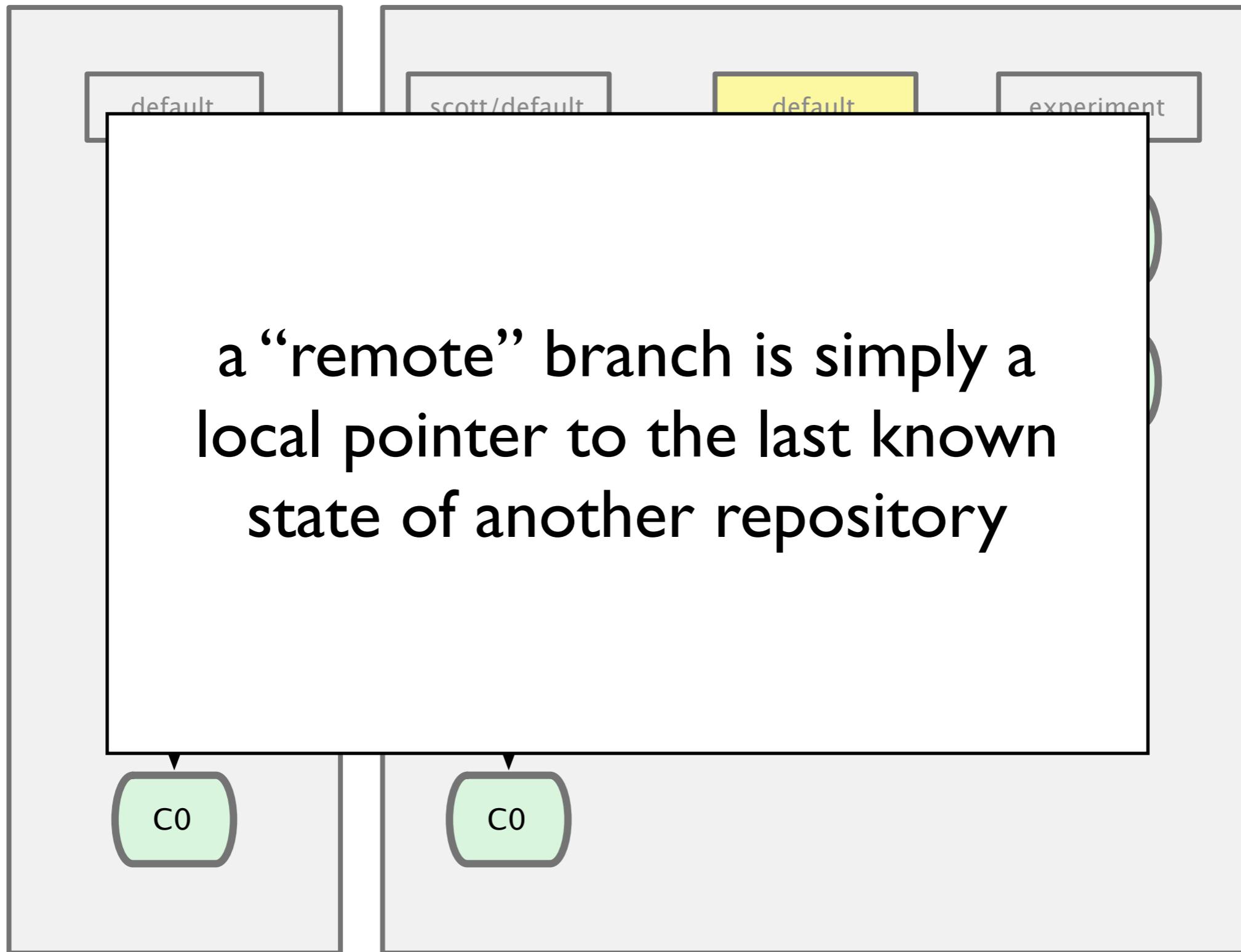


jessica



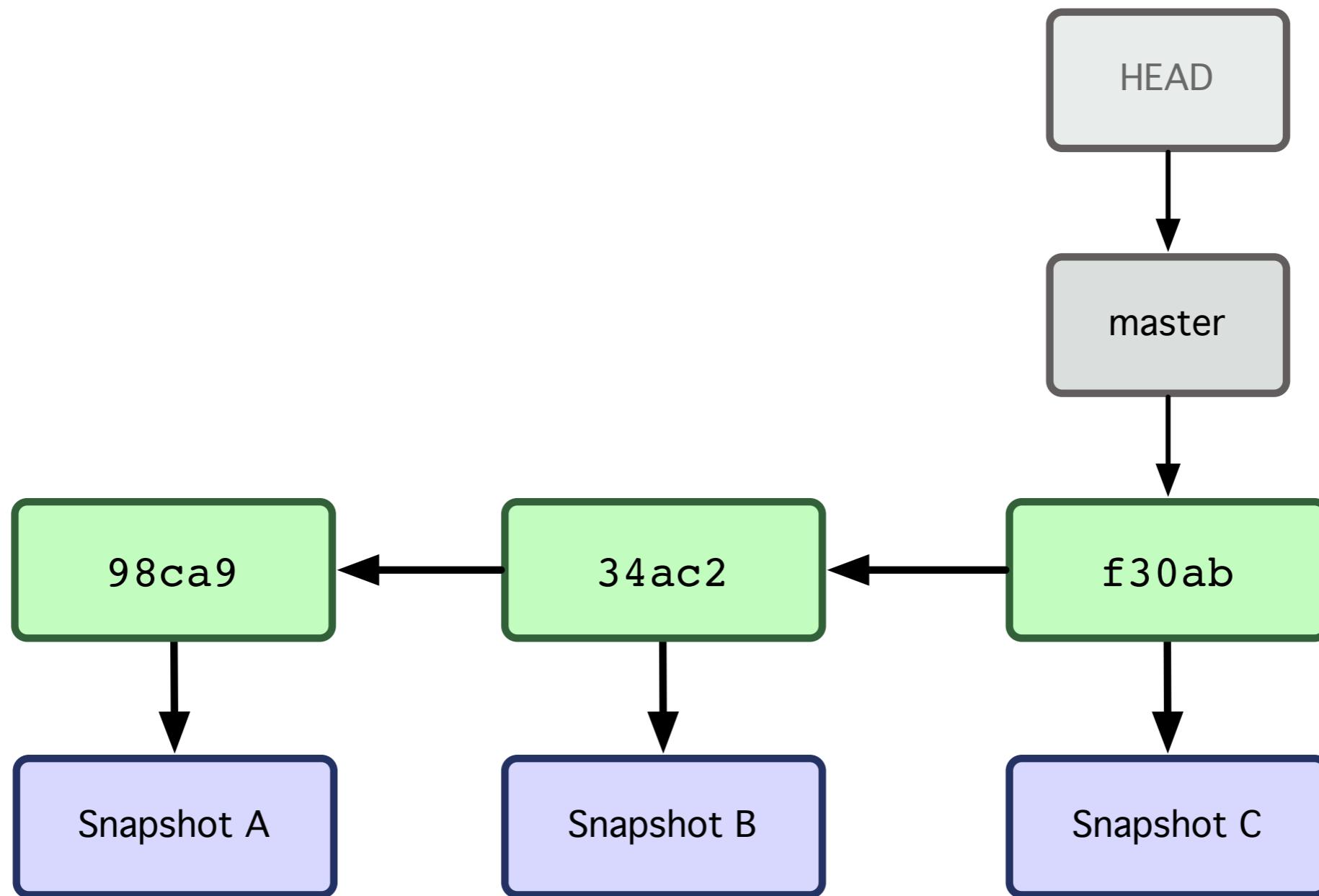
scott

jessica

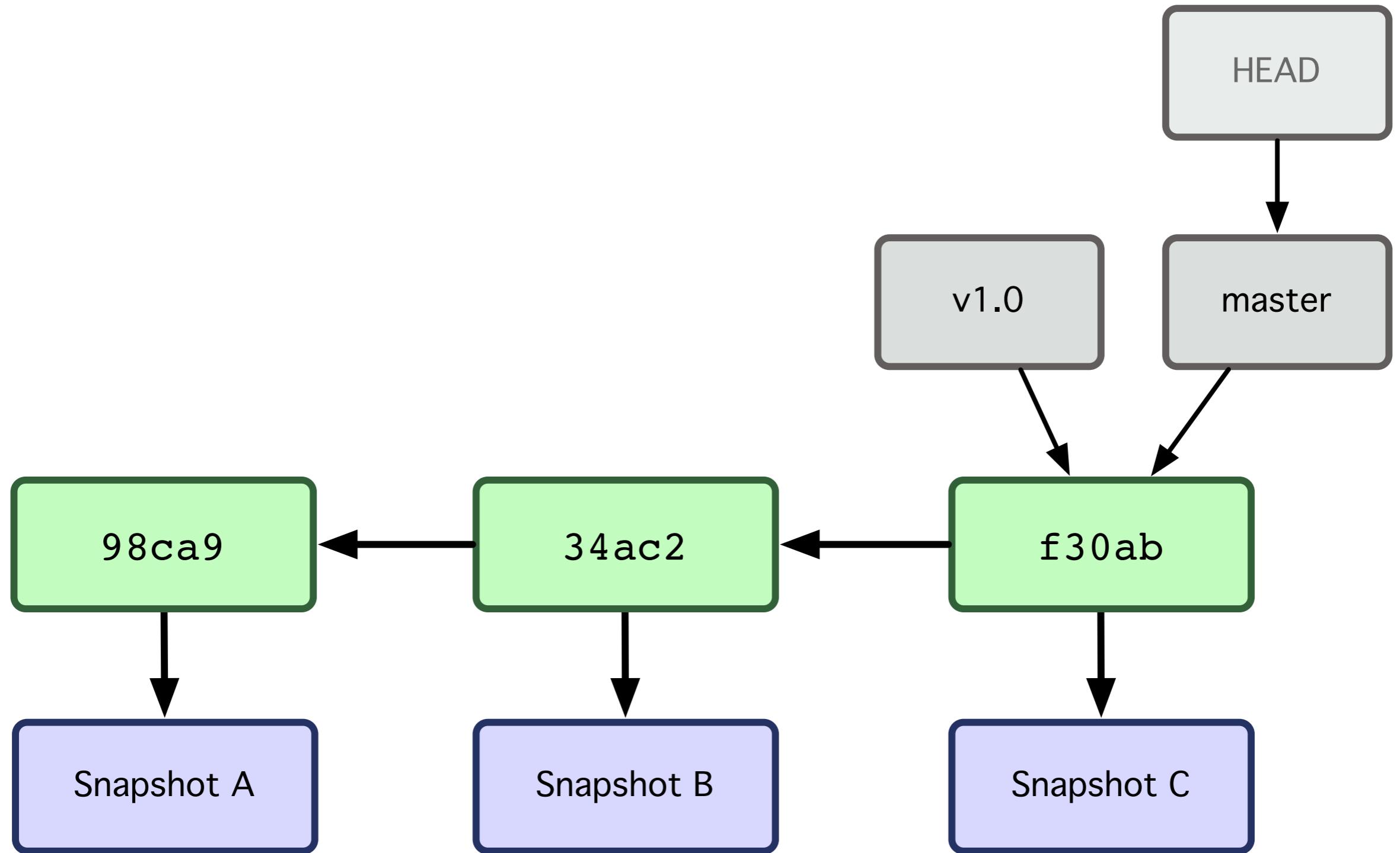


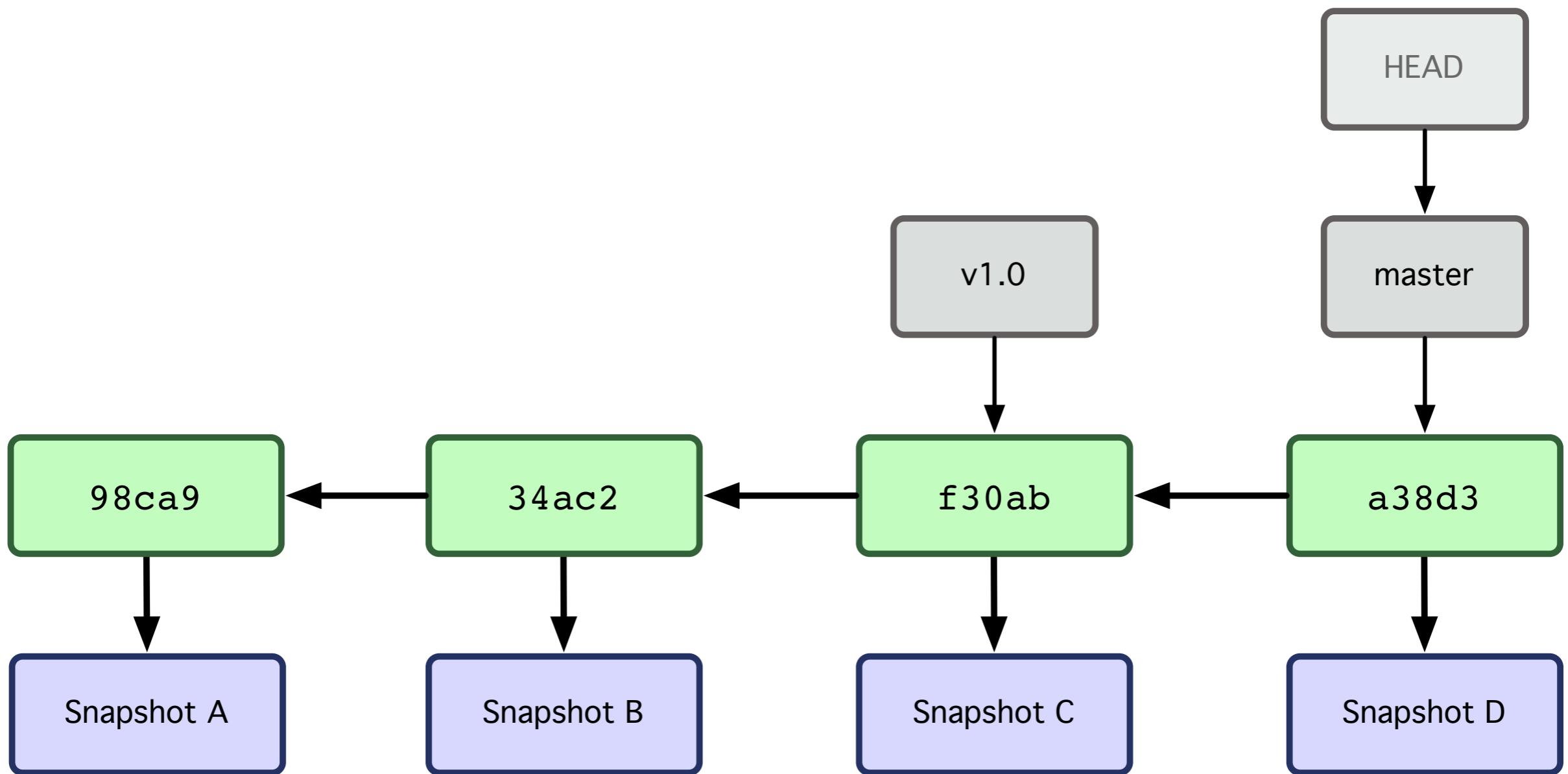
Tagging

lightweight tags



git tag v1.0



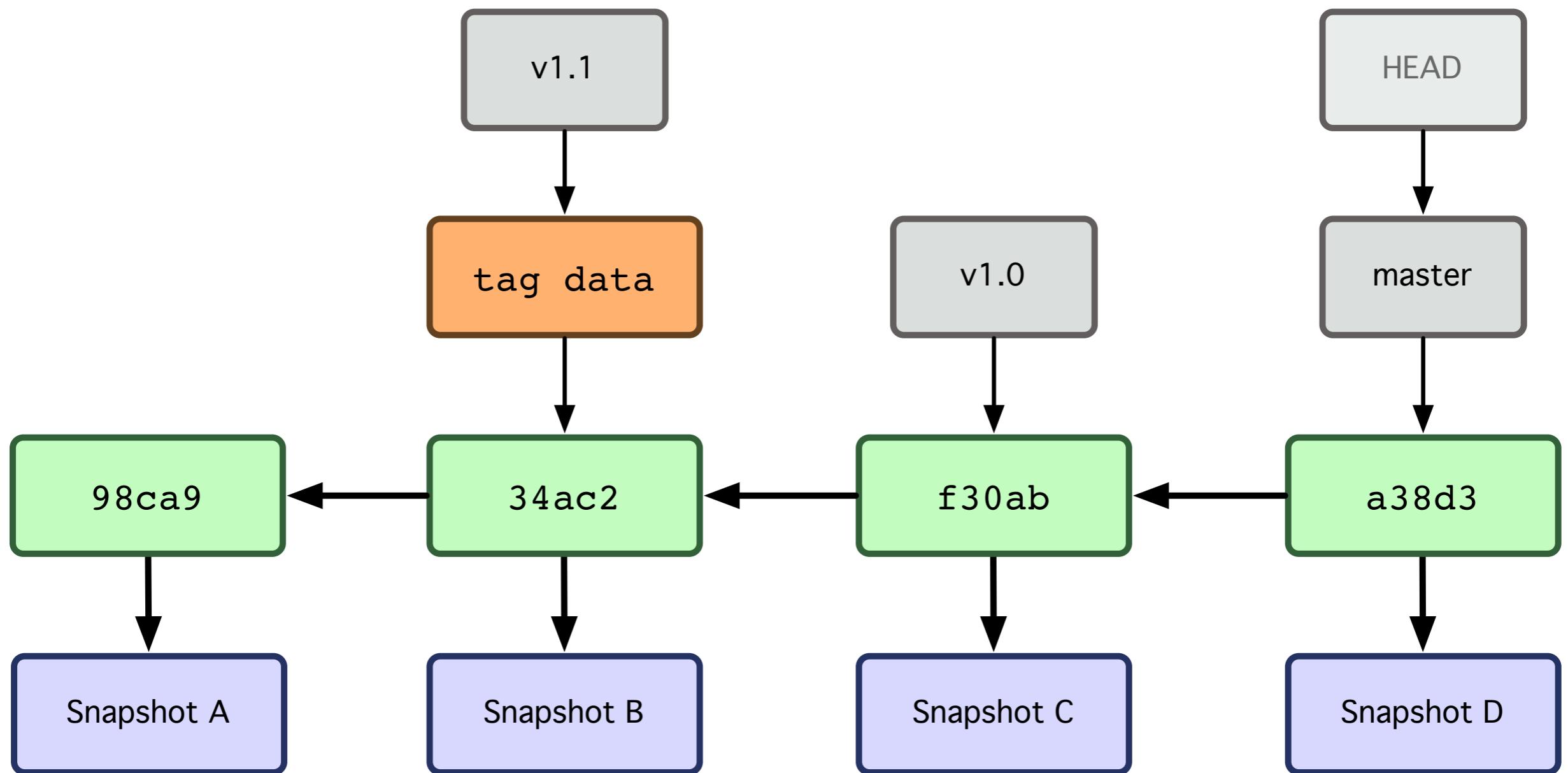


annotated tags

```
git tag -a v1.1 34ac2
```

49e11..

tag	size
object	ae668
type	commit
tagger	Scott
tag	v1.3
my tag message that explains this tag	



tagging objects

```
git hash-object -w file.txt
```

```
$ git hash-object -w ~/README.txt  
35a0dd3af166e09ac378dfeb95953923e71ea45b
```

```
$ git hash-object -w ~/README.txt  
35a0dd3af166e09ac378dfeb95953923e71ea45b
```

```
$ git tag -a my_readme 35a0dd3af166e09ac378dfe
```

```
$ git hash-object -w ~/README.txt  
35a0dd3af166e09ac378dfeb95953923e71ea45b
```

```
$ git tag -a my_readme 35a0dd3af166e09ac378dfe
```

```
$ git show my_readme  
tag my_readme  
Tagger: Scott Chacon <schacon@gmail.com>  
Date: Tue Mar 16 08:36:24 2010 -0700
```

my readme
My README File
by Scott Chacon

This is my README

sharing tags

git push --tags

Git Inspection

Revision Selection

Revision Selection

alternate ways to refer to
objects or ranges of objects

Revision Selection

full sha- l

partial sha- l

branch or tag name

caret parent

tilde spec

blob spec

relative specs

ranges

Full SHA1

6e453f523fa1da50ecb04431101112b3611c6a4d

Partial SHA1

6e453f523fa1da50ecb04431101112b3611c6a4d

6e453f523fa1da50

6e453

Branch, Remote or Tag Name

v1.0

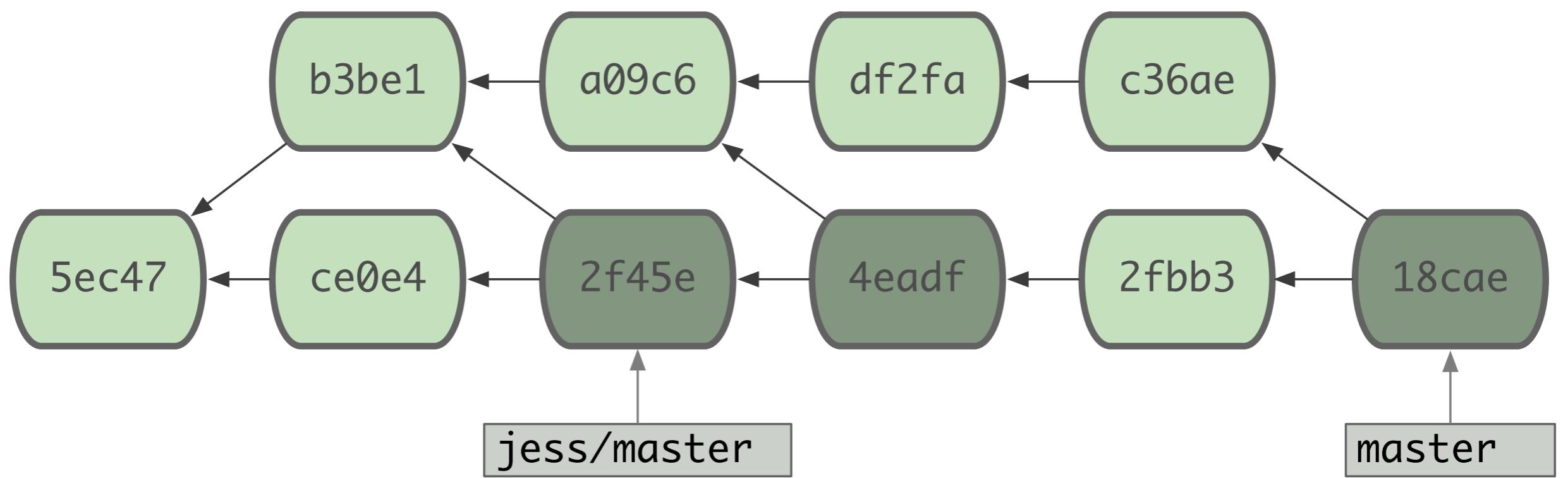
default

m/cupcake

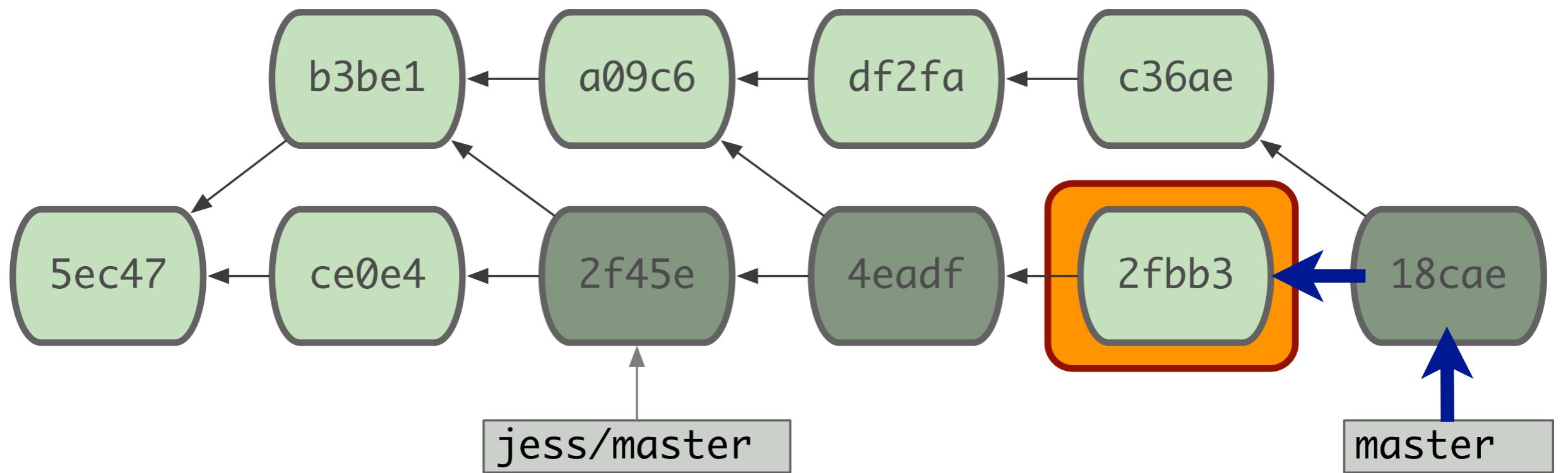
Caret Parent

default^{^2}

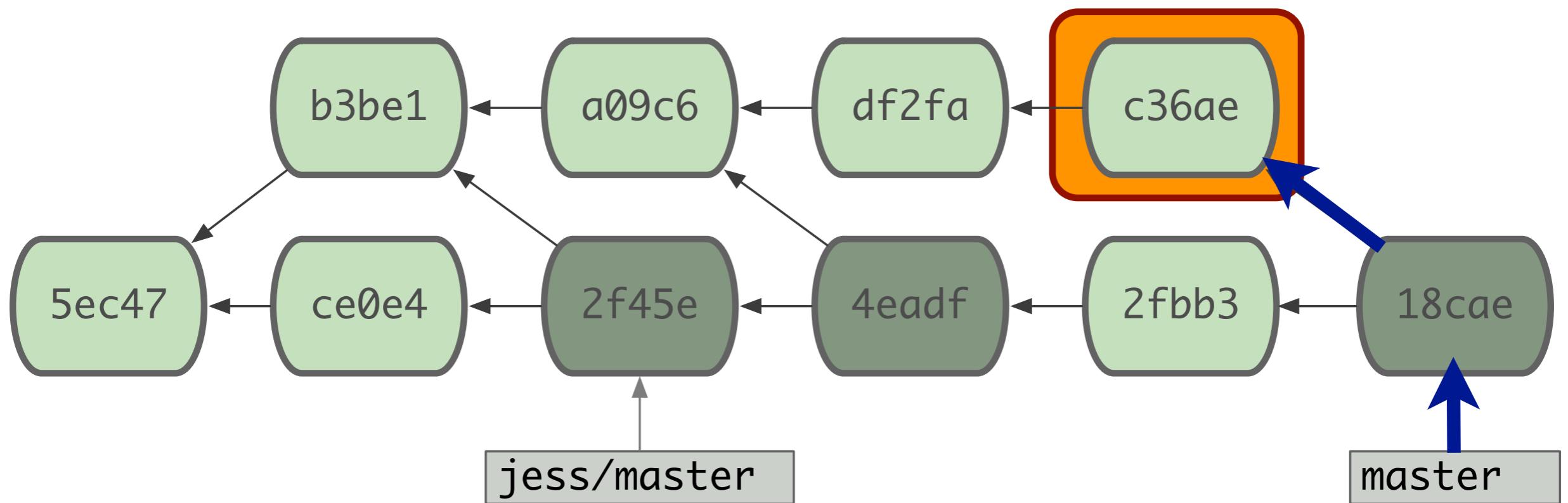
2nd parent of ‘default’



master^Λ



master^{^2}

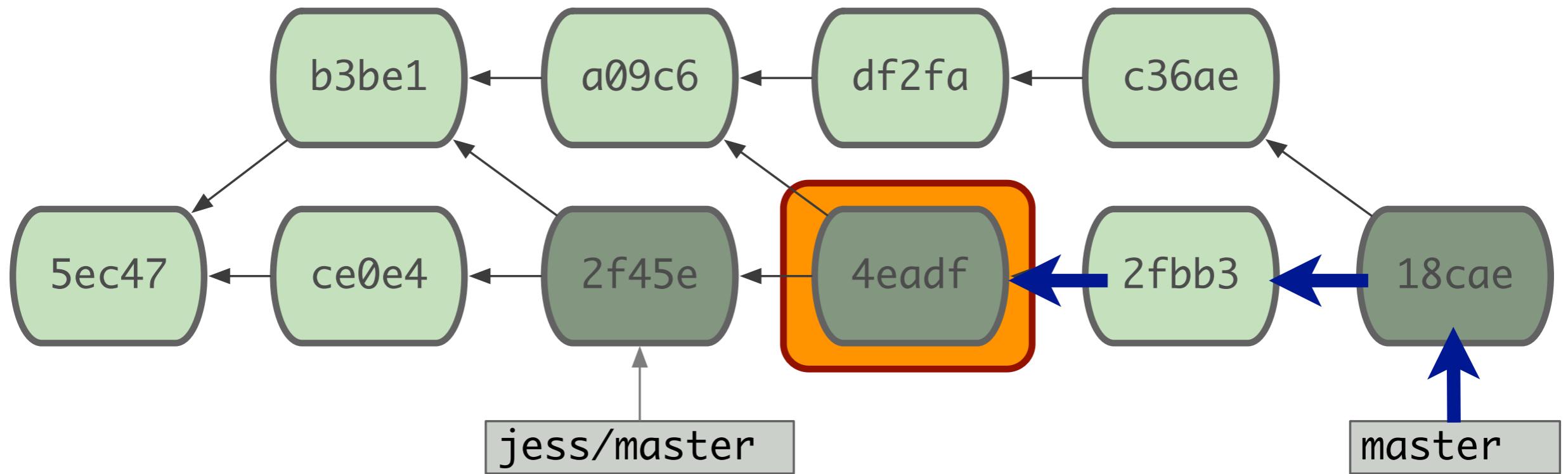


Tilde Spec

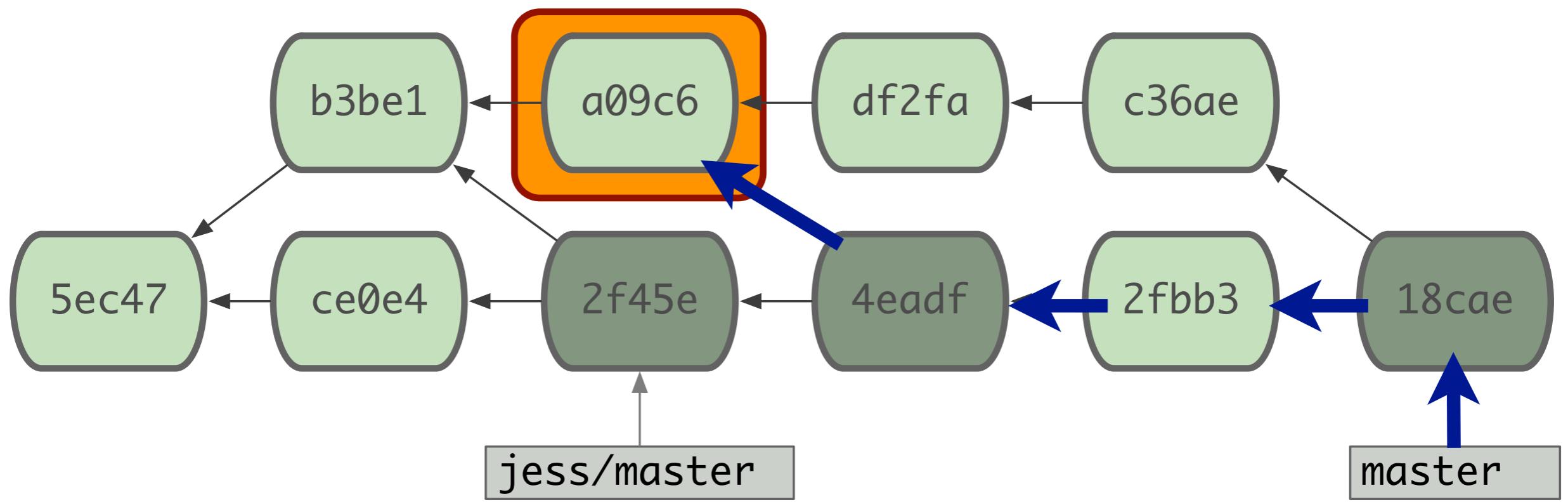
default~2

grandparent of ‘default’
(parent of the parent)

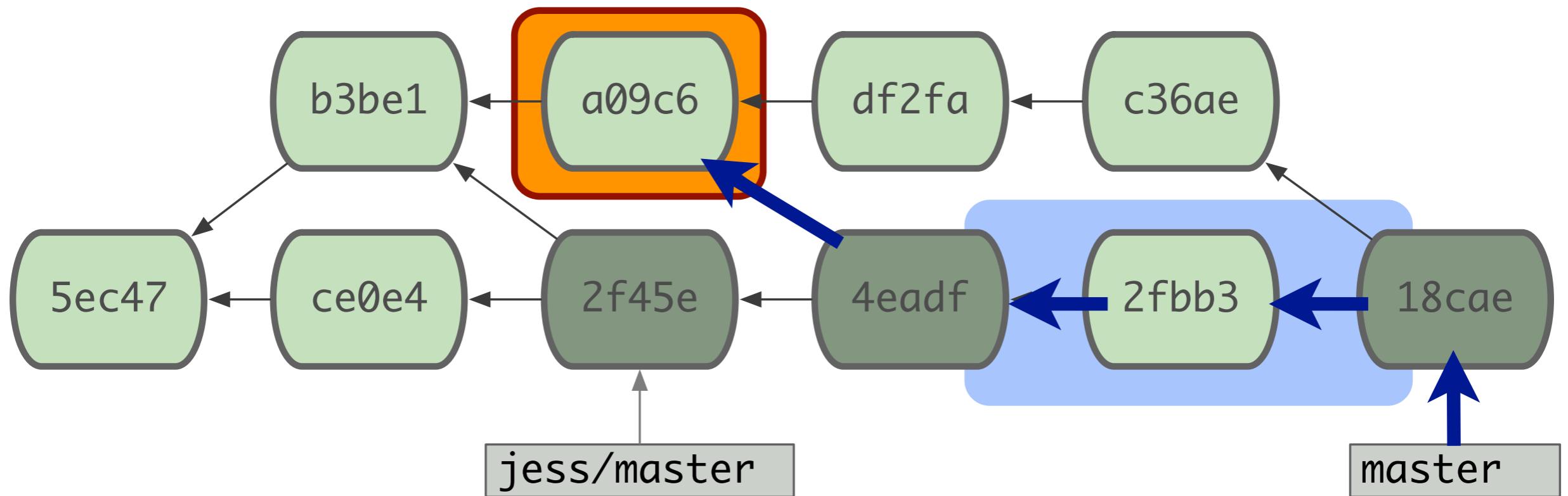
master~2



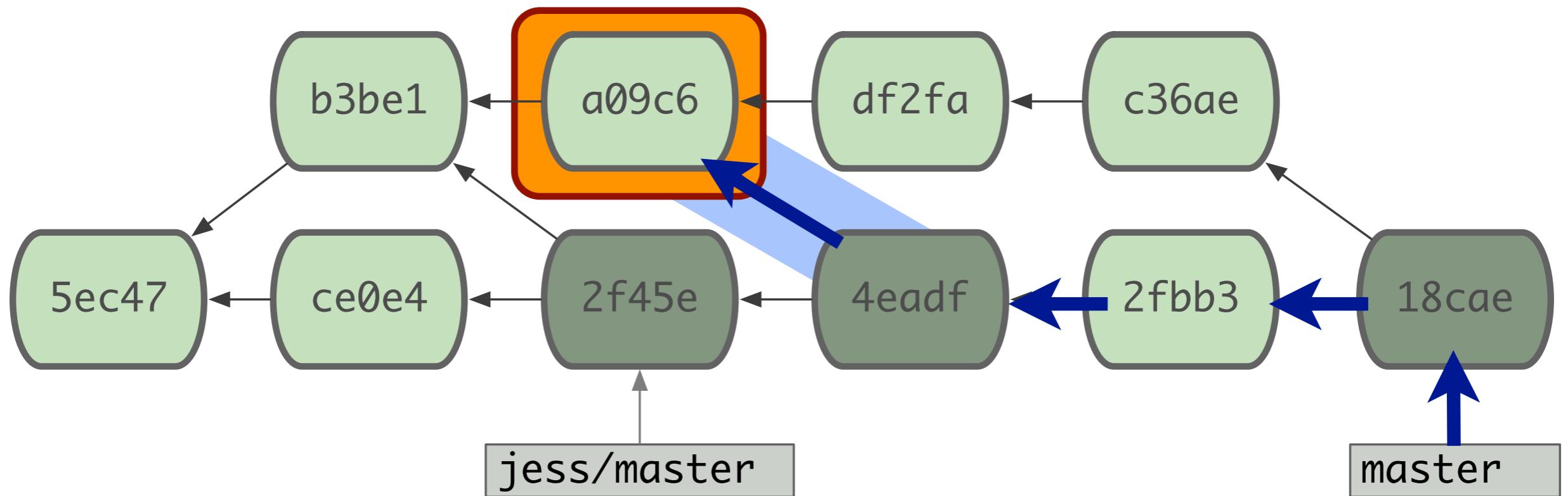
master^{^2} master~2^2



master^{^2}
master~2^2



master^{^2}
master~2^{^2}



Blob Spec

default:path/to/file

blob of that file in ‘default’ commit

Relative Specs

master@{yesterday}

the commit ‘master’ was at yesterday

Relative Specs

master@{5}

the 5th prior value of ‘master’ (locally)

Ranges

`ce0e4 .. e4272`

every commit reachable by e4272
that is not reachable by ce034

Ranges

[old] .. [new]

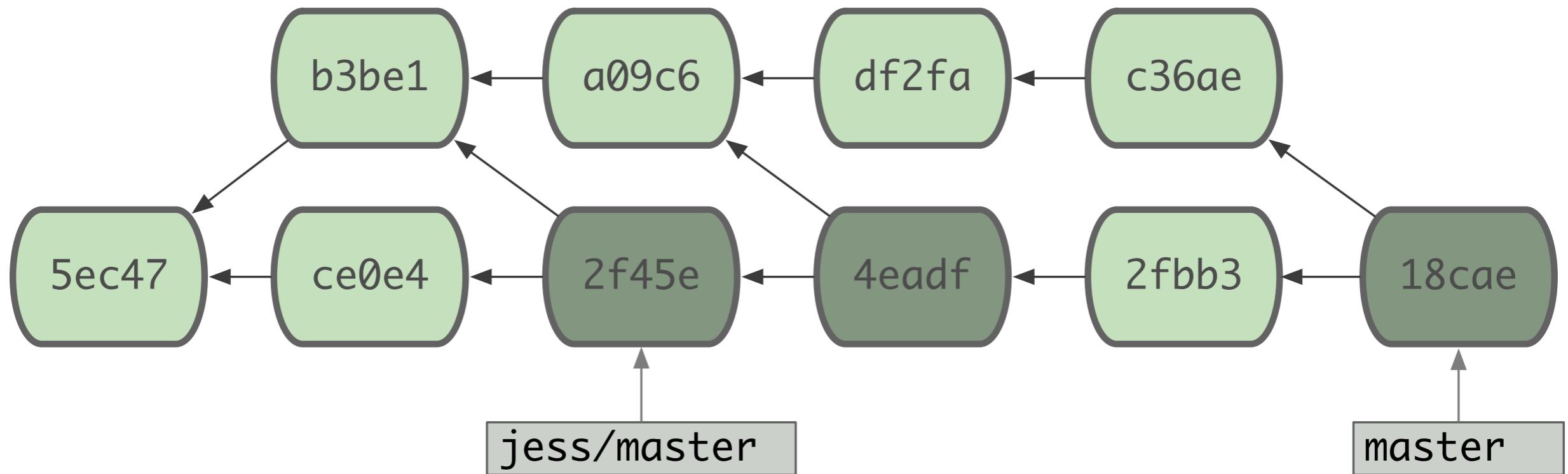
every commit reachable by [new]
that is not reachable by [old]

Ranges

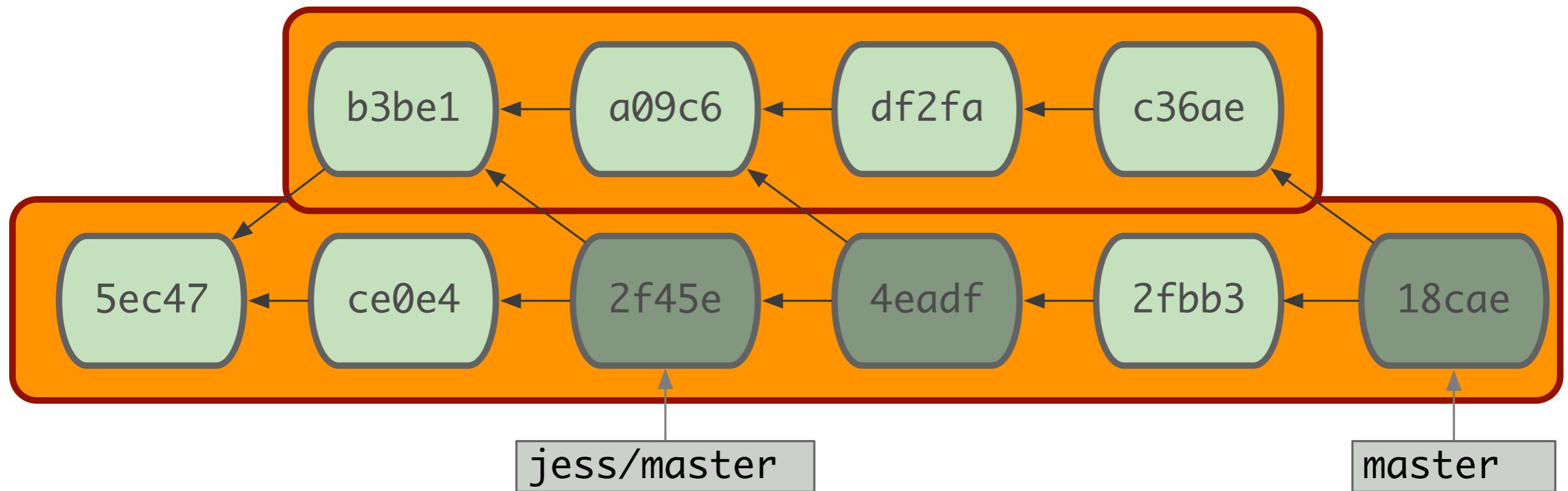
ce0e4 ..

everything since a commit

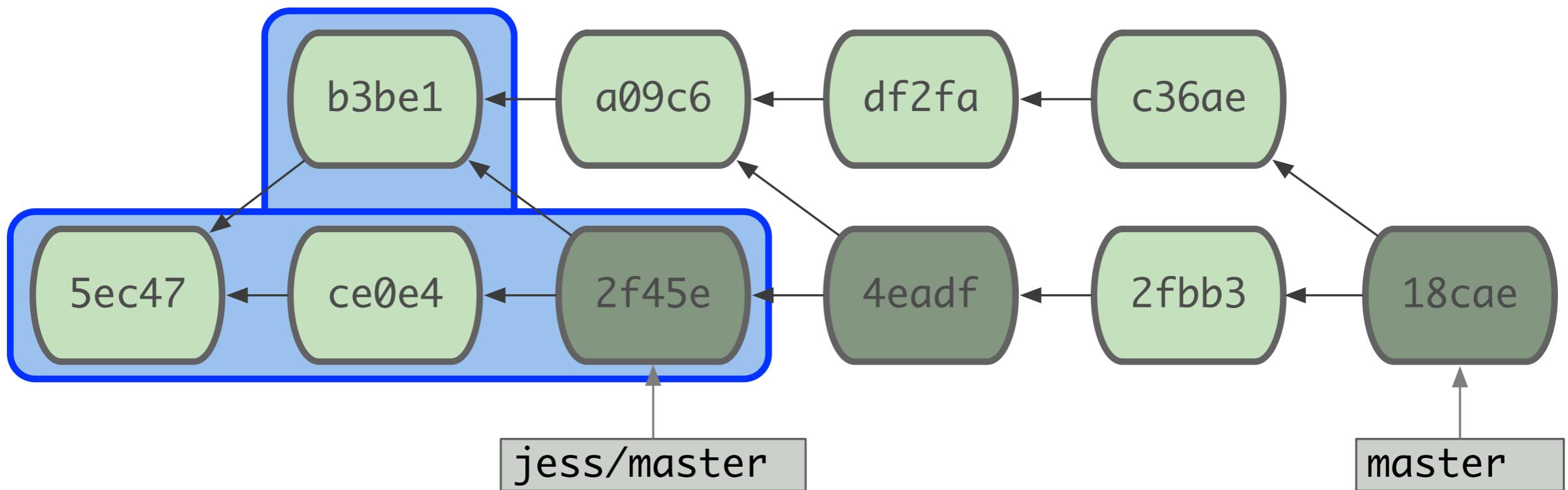
jess/master.master



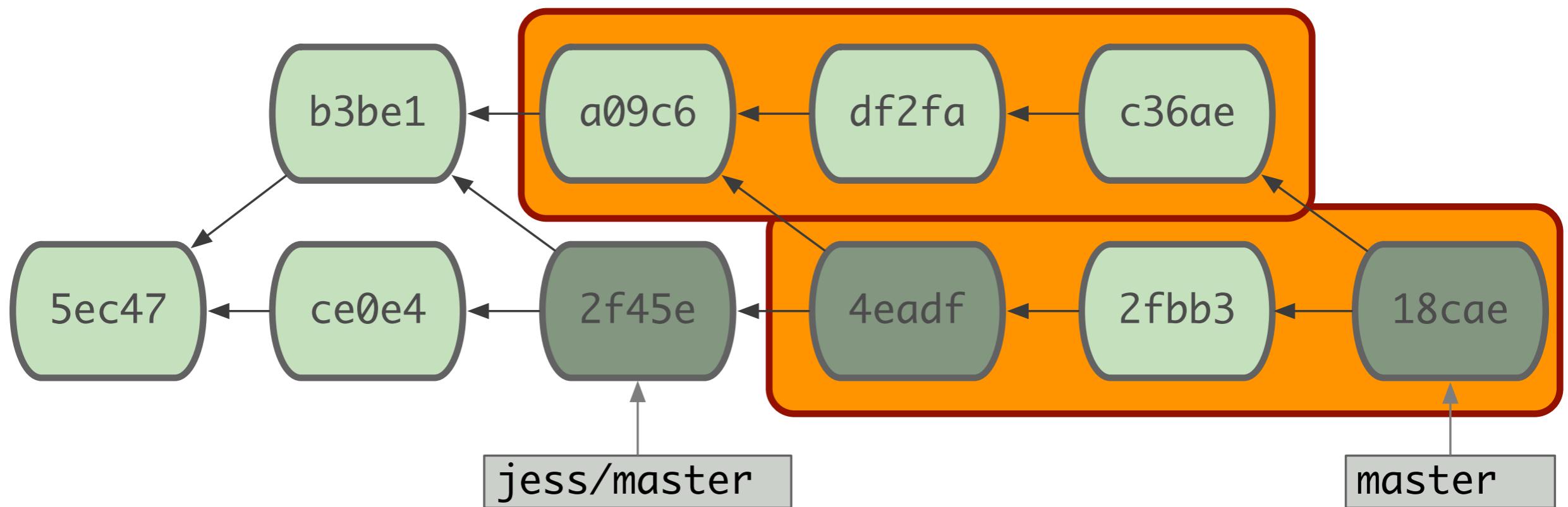
jess/master..master



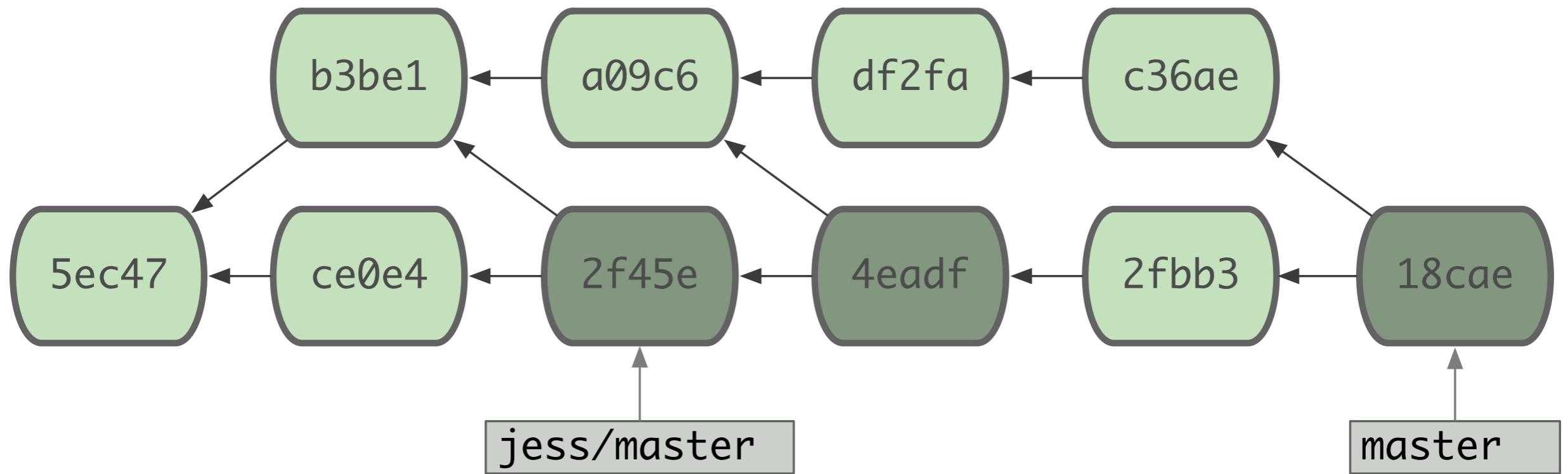
jess/master.master



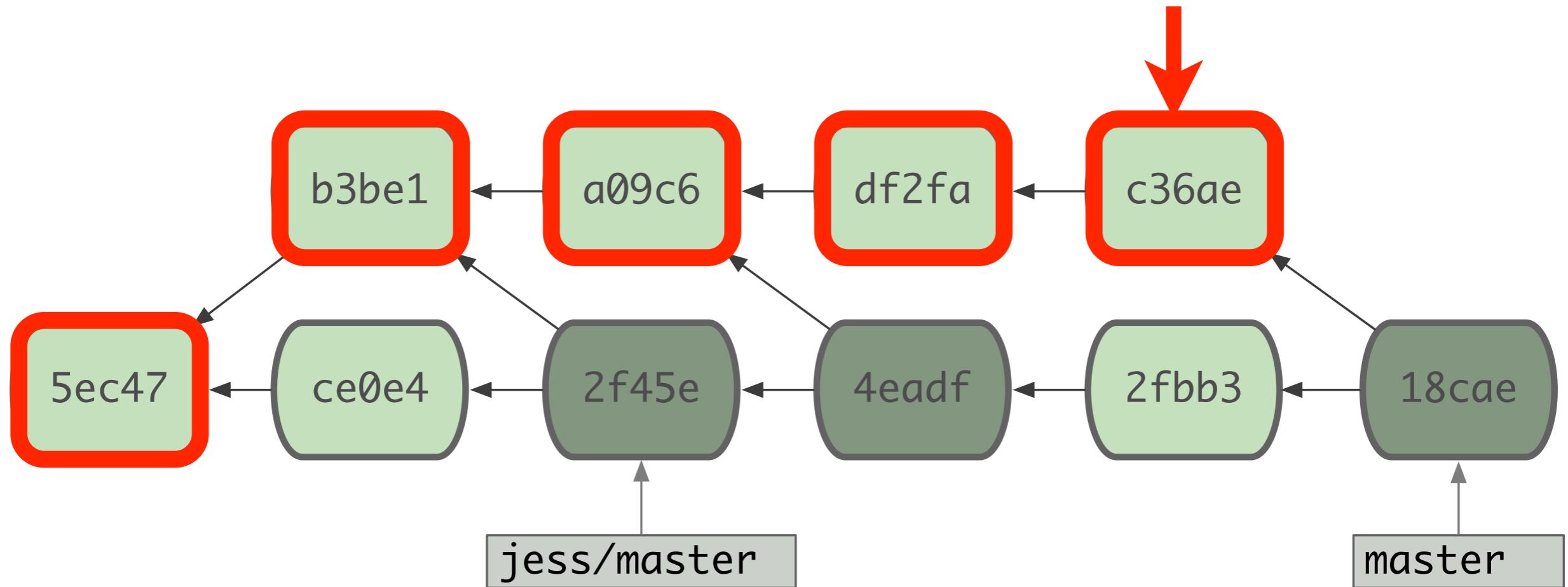
jess/master..master



jess/master..c36ae

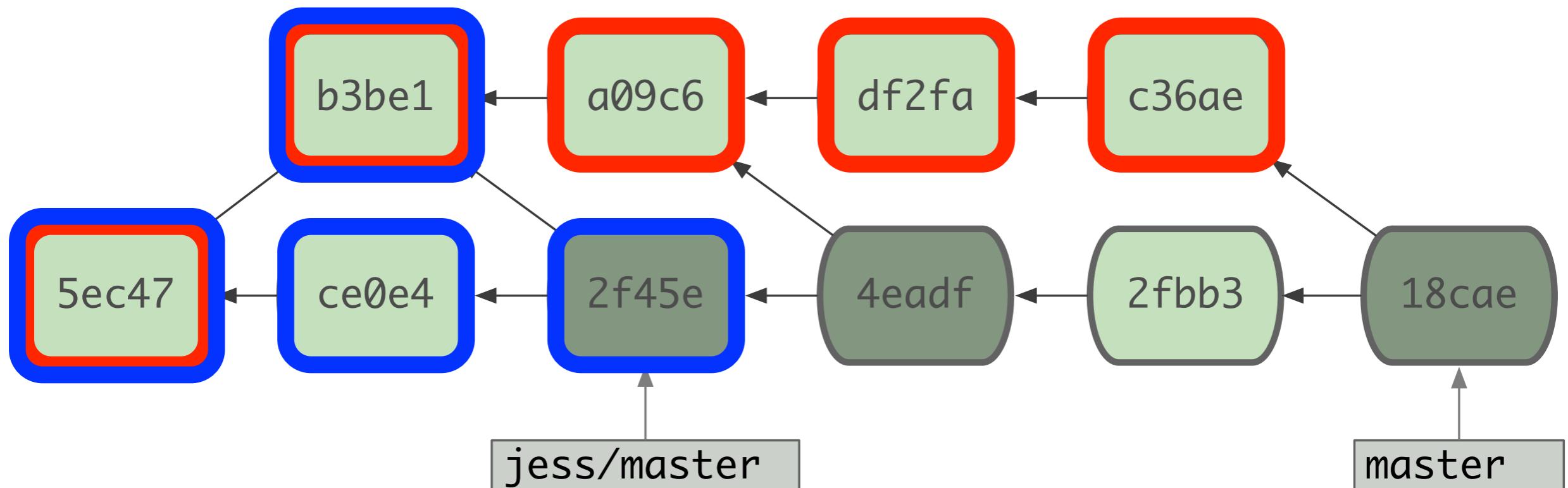


jess/master..c36ae

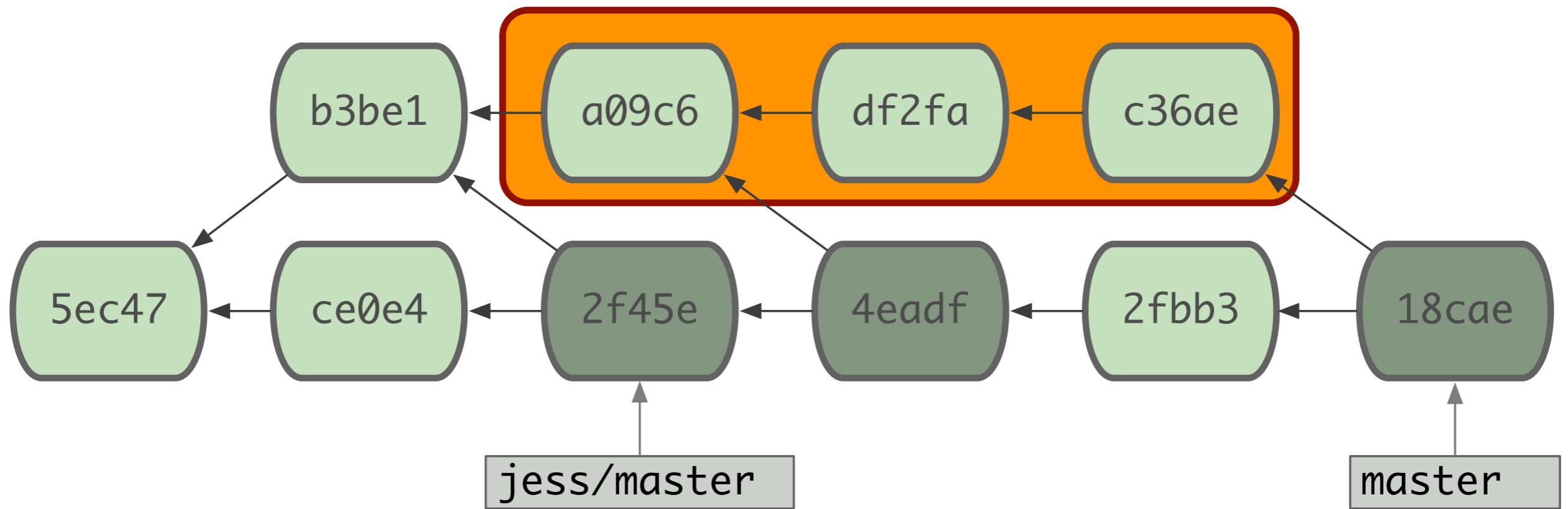




jess/master..c36ae



jess/master..c36ae

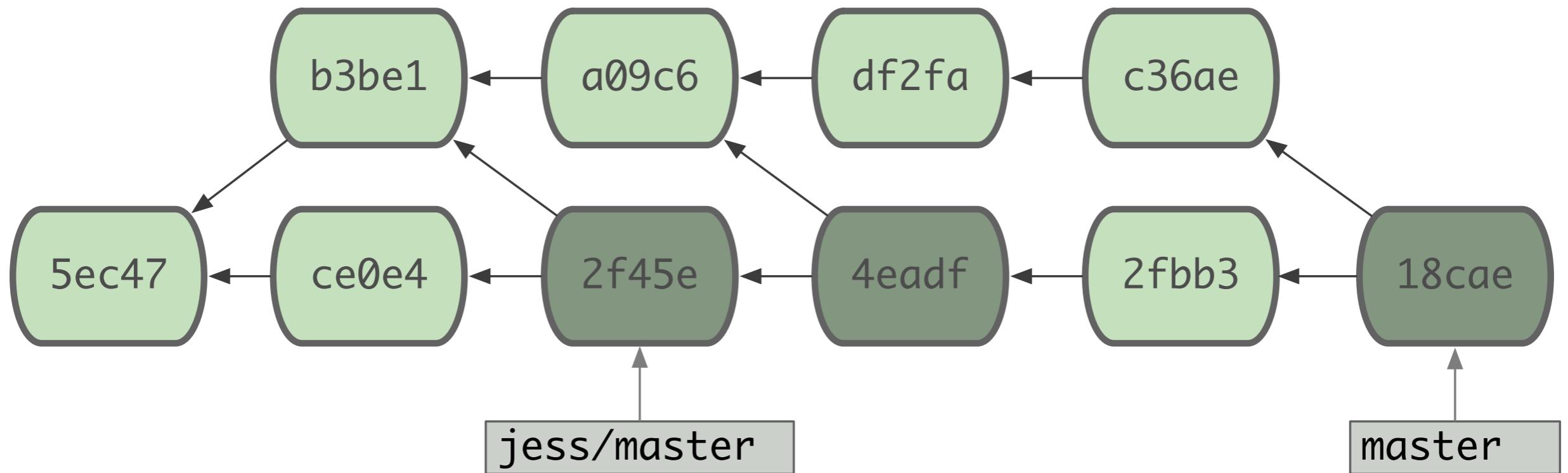


Ranges

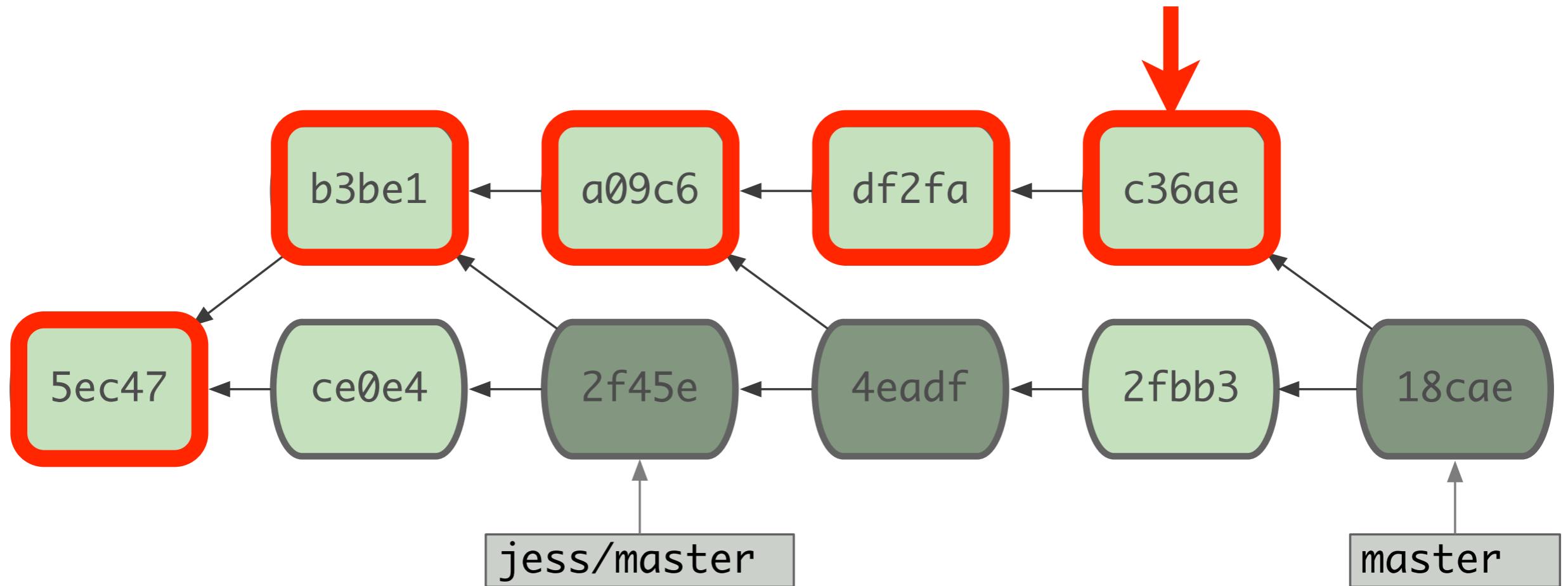
ce0e4 . . . e4272

every commit reachable by either
but not commits reachable by both

jess/master...c36ae

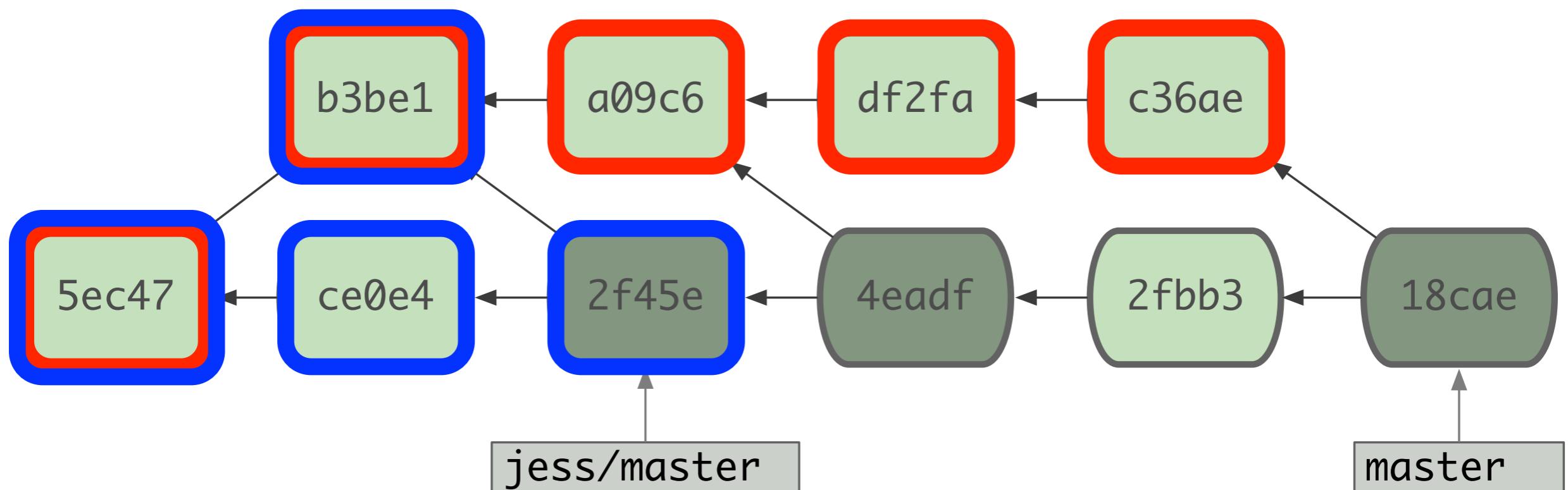


jess/master...c36ae

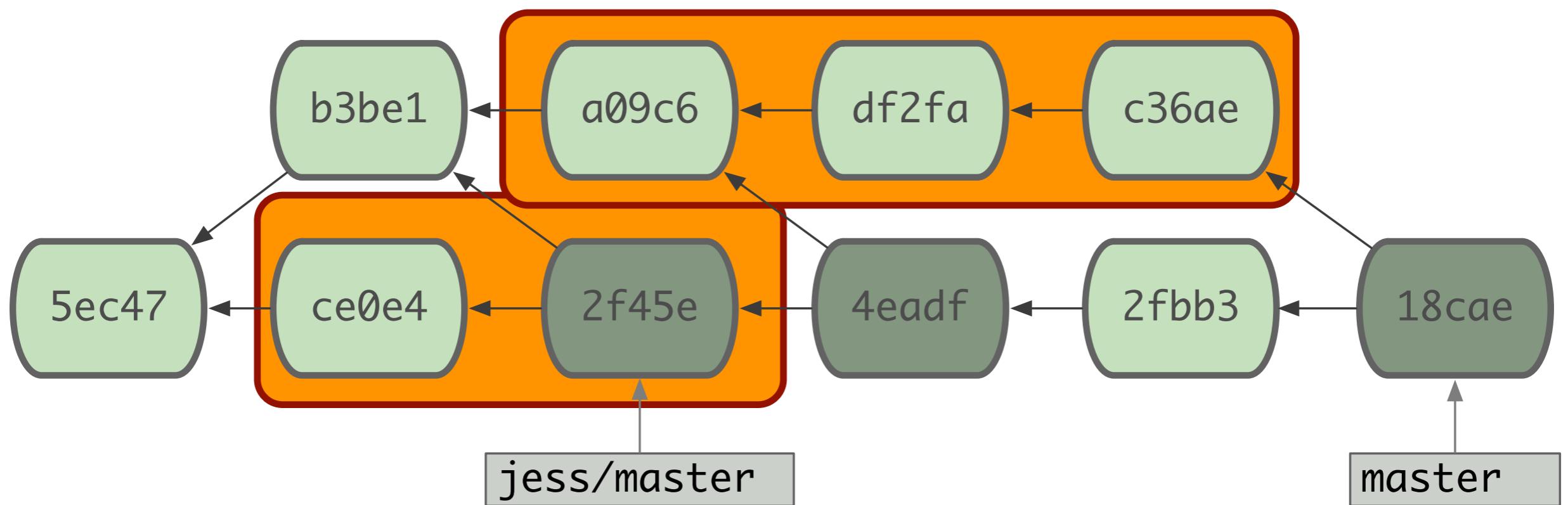




jess/master...c36ae



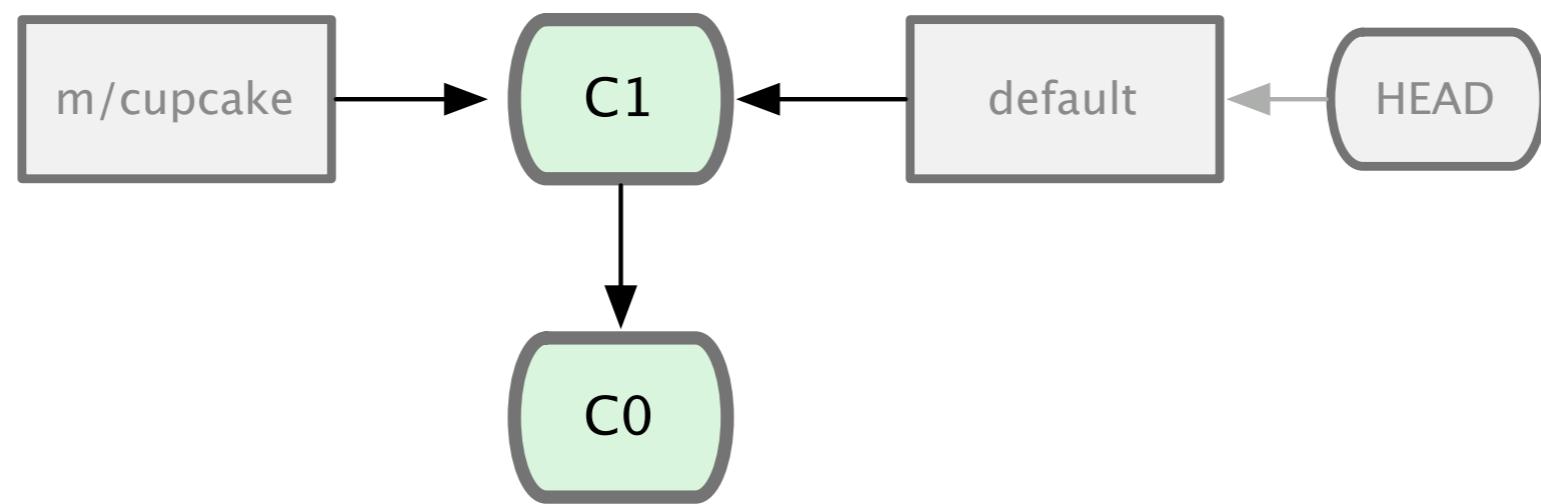
jess/master...c36ae

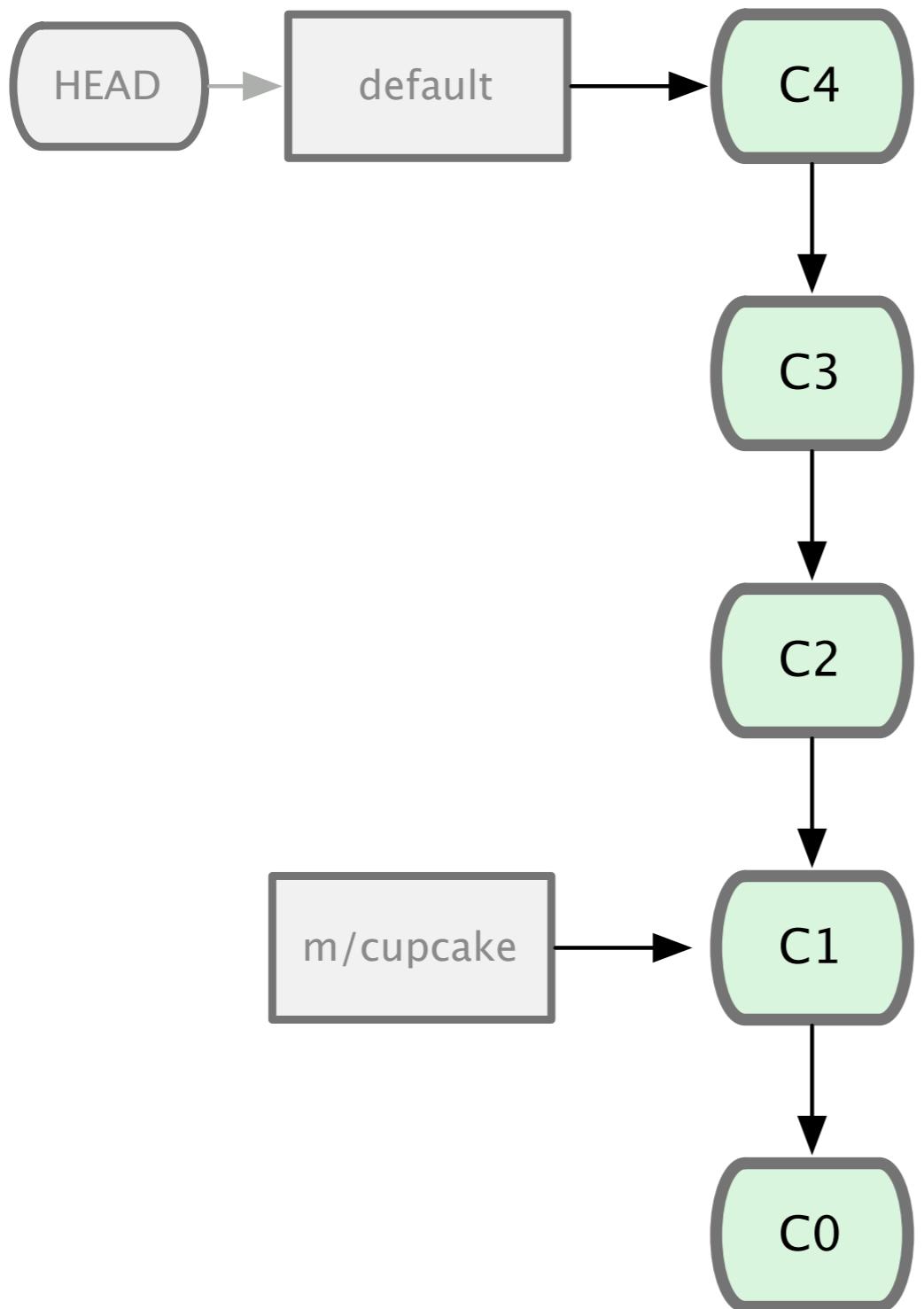


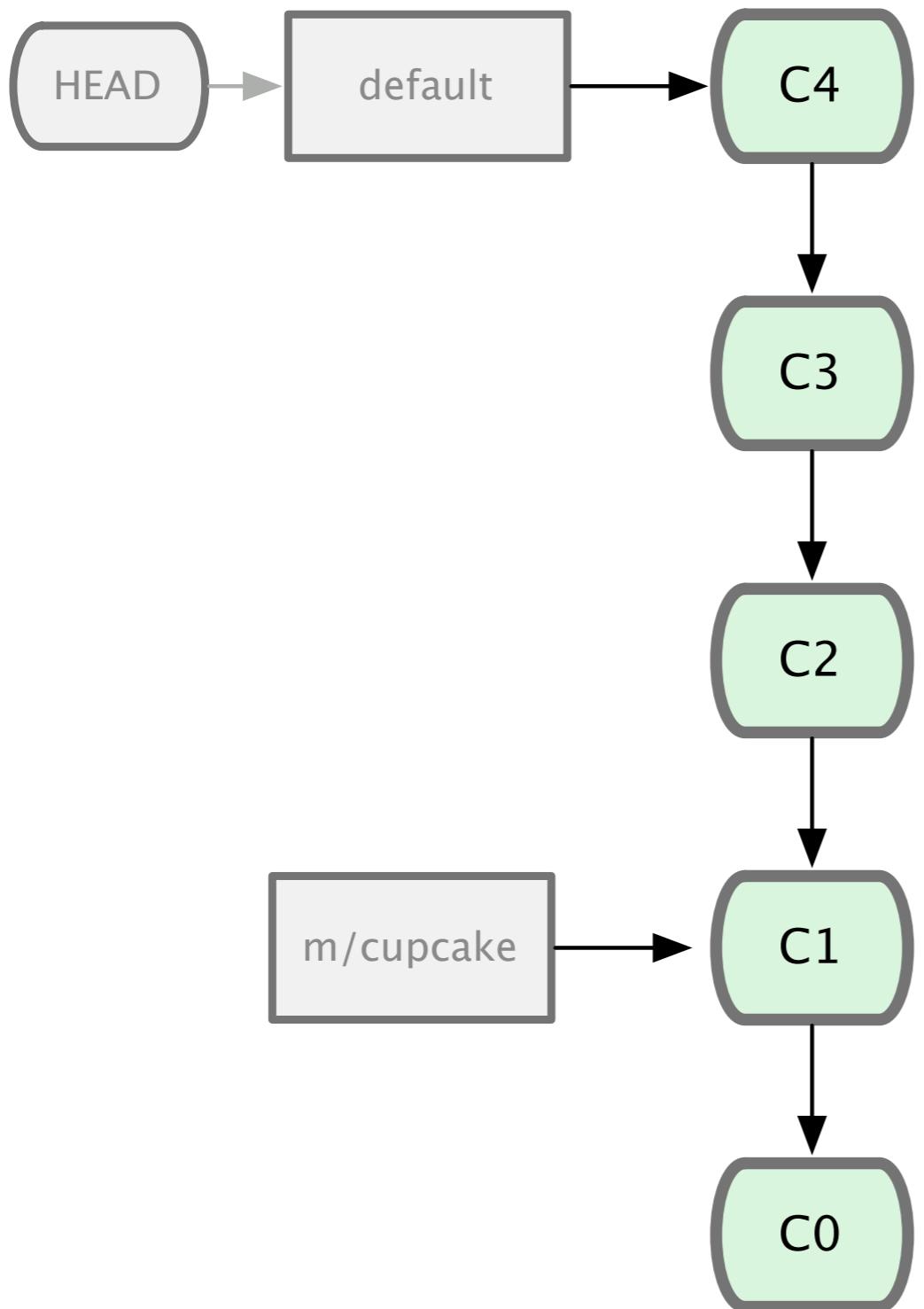
Advanced Log

Log Subsets

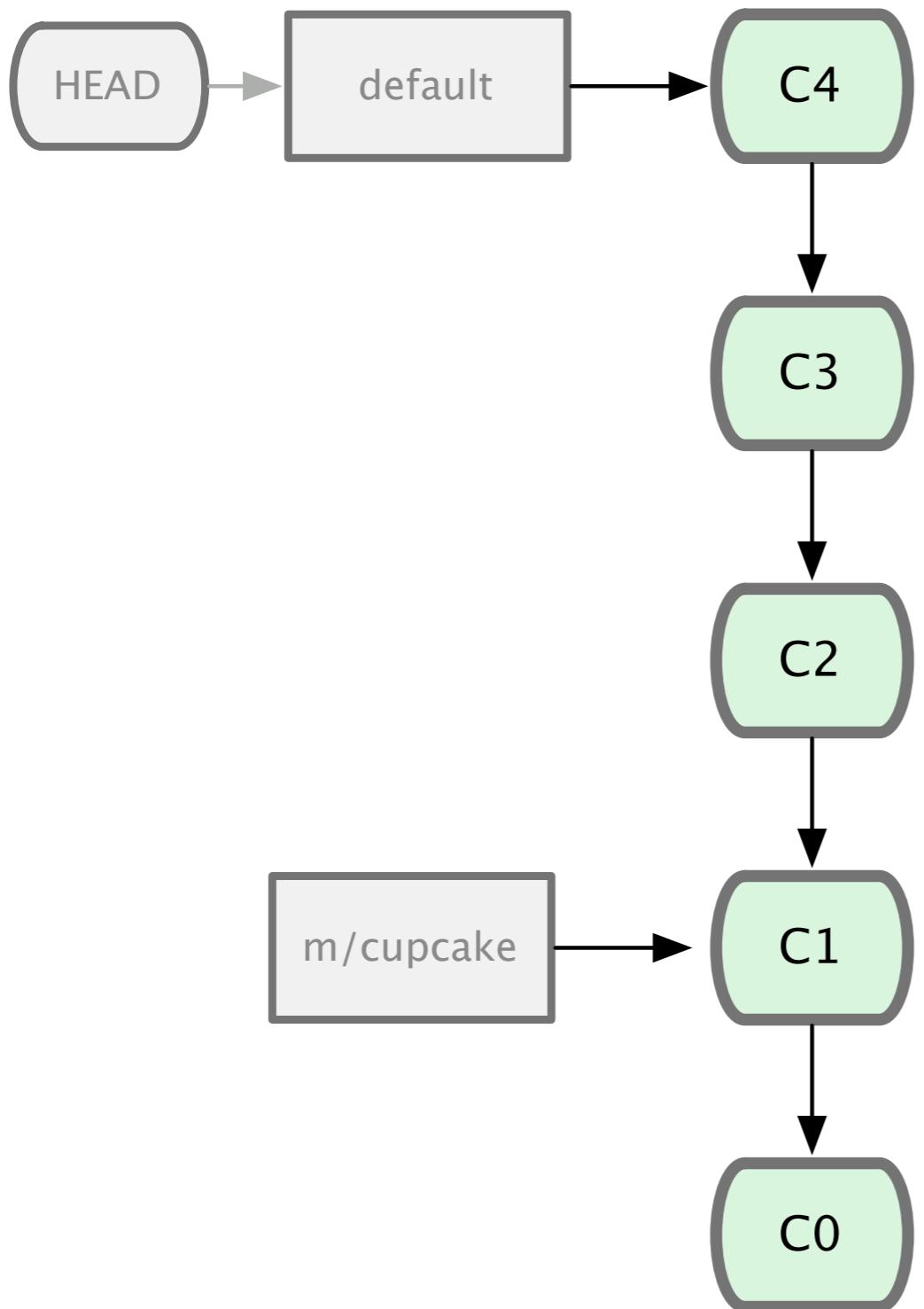
```
git log m/cupcake..
```



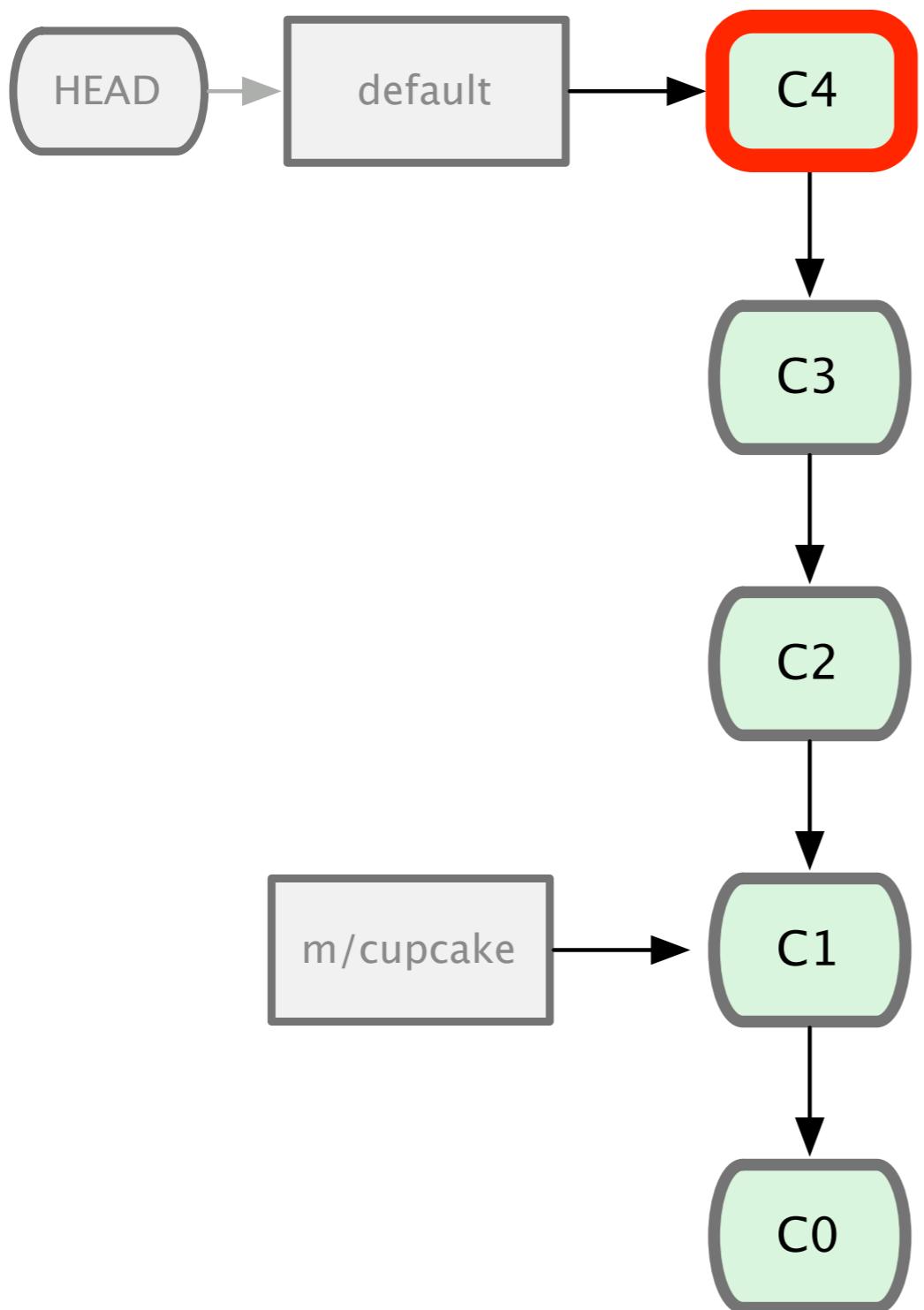




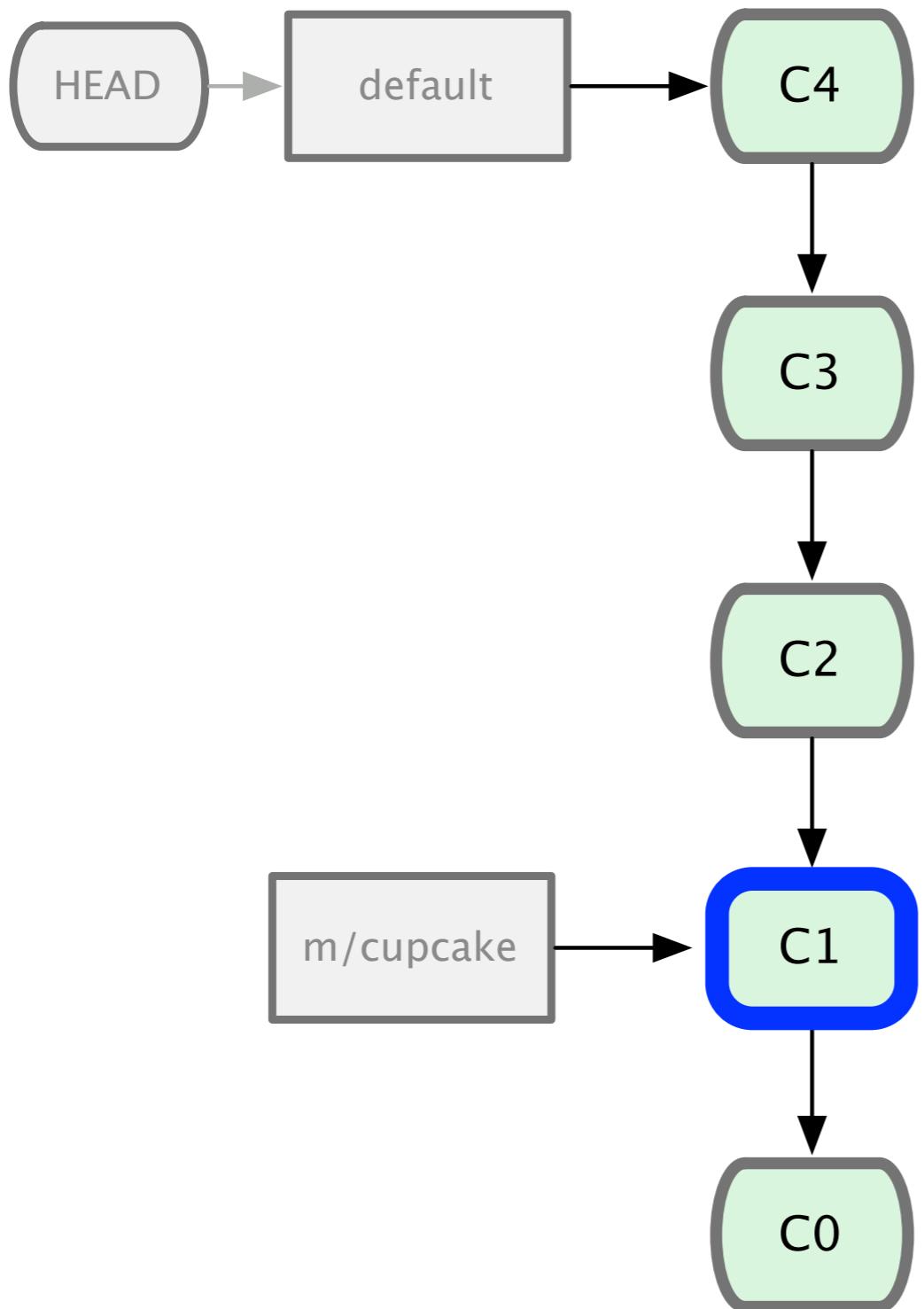
git log m/cupcake..



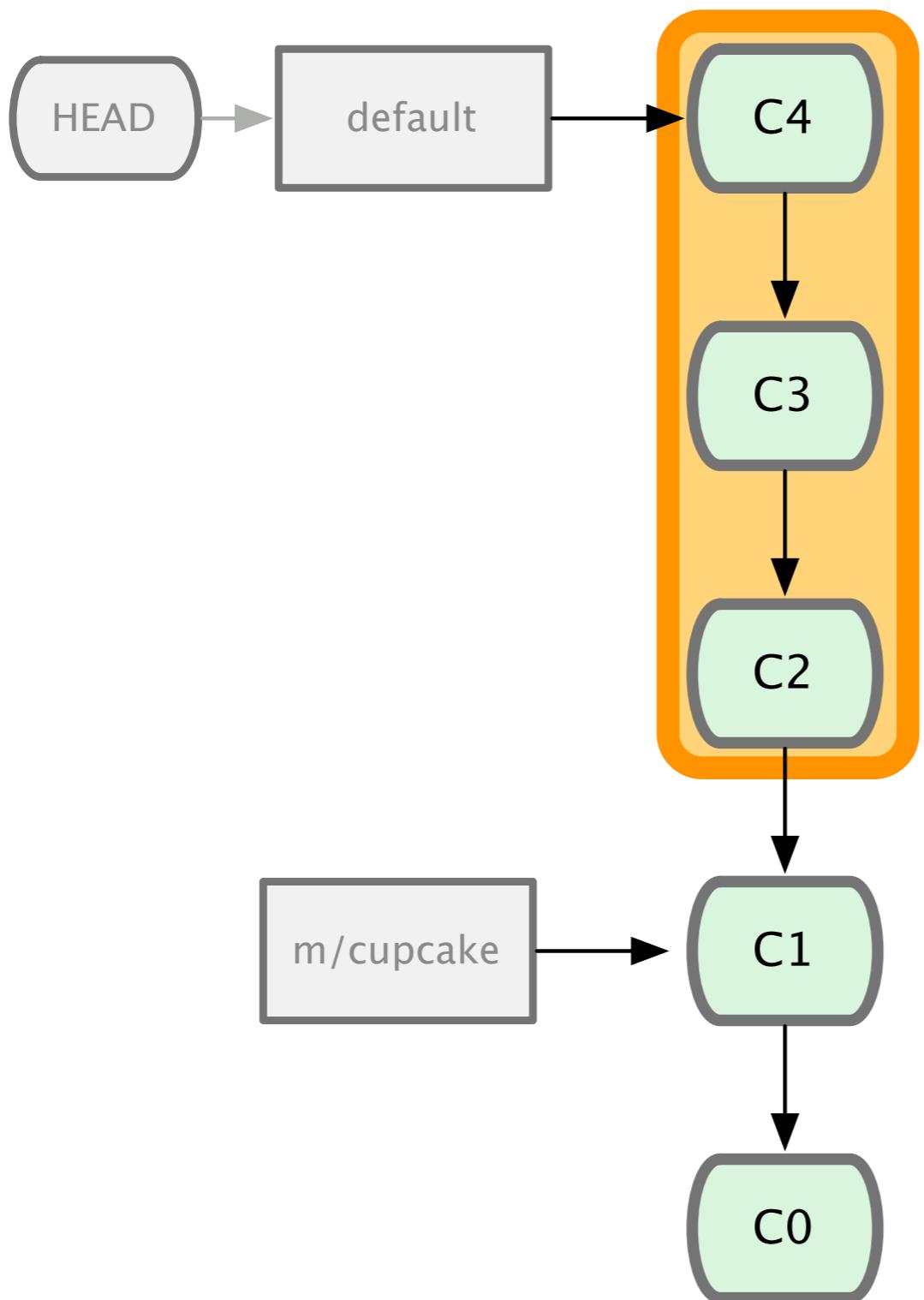
git log m/cupcake..HEAD



git log m/cupcake..HEAD



git log **m/cupcake**..HEAD



git log m/cupcake..HEAD

commit 72d404debaa804fca82fd9cf710fbde48c7305c6

Author: Shawn O. Pearce <spearce@spearce.org>

Date: Sun Oct 12 13:13:59 2008 -0700

test-lib: fix broken printf

b8eecaf888d219633f4c29e8b6a90fc21a46dfd introduced usage of
printf without a format string.

Signed-off-by: Shawn O. Pearce <spearce@spearce.org>

commit 969c877506cf8cc760c7b251fef6c5b6850bfc19

Author: Jeff King <peff@peff.net>

Date: Sun Oct 12 00:06:11 2008 -0400

git apply --directory broken for new files

We carefully verify that the input to git-apply is sane,
including cross-checking that the filenames we see in "+++"
headers match what was provided on the command line of "diff
--git". When --directory is used, however, we ended up
comparing the unadorned name to one with the prepended root,
causing us to complain about a mismatch.

We simply need to prepend the root directory, if any, when
pulling the name out of the git header.

commit ff74126c03a8dfd04e7533573a5c420f2a7112ac

Author: Johannes Schindelin <Johannes.Schindelin@gmx.de>

Date: Fri Oct 10 13:42:12 2008 +0200

rebase -i: do not fail when there is no commit to cherry-pick

C4

commit 72d404debaa804fca82fd9cf710fbde48c7305c6
Author: Shawn O. Pearce <spearce@spearce.org>
Date: Sun Oct 12 13:13:59 2008 -0700

test-lib: fix broken printf

b8eecaf888d219633f4c29e8b6a90fc21a46dfd introduced usage of printf without a format string.

Signed-off-by: Shawn O. Pearce <spearce@spearce.org>

C3

commit 969c877506cf8cc760c7b251fef6c5b6850bfc19
Author: Jeff King <peff@peff.net>
Date: Sun Oct 12 00:06:11 2008 -0400

git apply --directory broken for new files

We carefully verify that the input to git-apply is sane, including cross-checking that the filenames we see in "+++" headers match what was provided on the command line of "diff --git". When --directory is used, however, we ended up comparing the unadorned name to one with the prepended root, causing us to complain about a mismatch.

We simply need to prepend the root directory, if any, when pulling the name out of the git header.

C2

commit ff74126c03a8dfd04e7533573a5c420f2a7112ac
Author: Johannes Schindelin <Johannes.Schindelin@gmx.de>
Date: Fri Oct 10 13:42:12 2008 +0200

rebase -i: do not fail when there is no commit to cherry-pick

```
git log m/cupcake..
```

```
git log m/cupcake..HEAD
```

```
git log HEAD ^m/cupcake
```

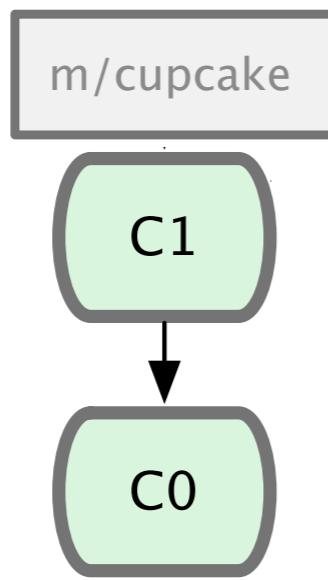
```
git log 35ae2 ^9fe42
```

```
git log 3e45a ^573eb
```

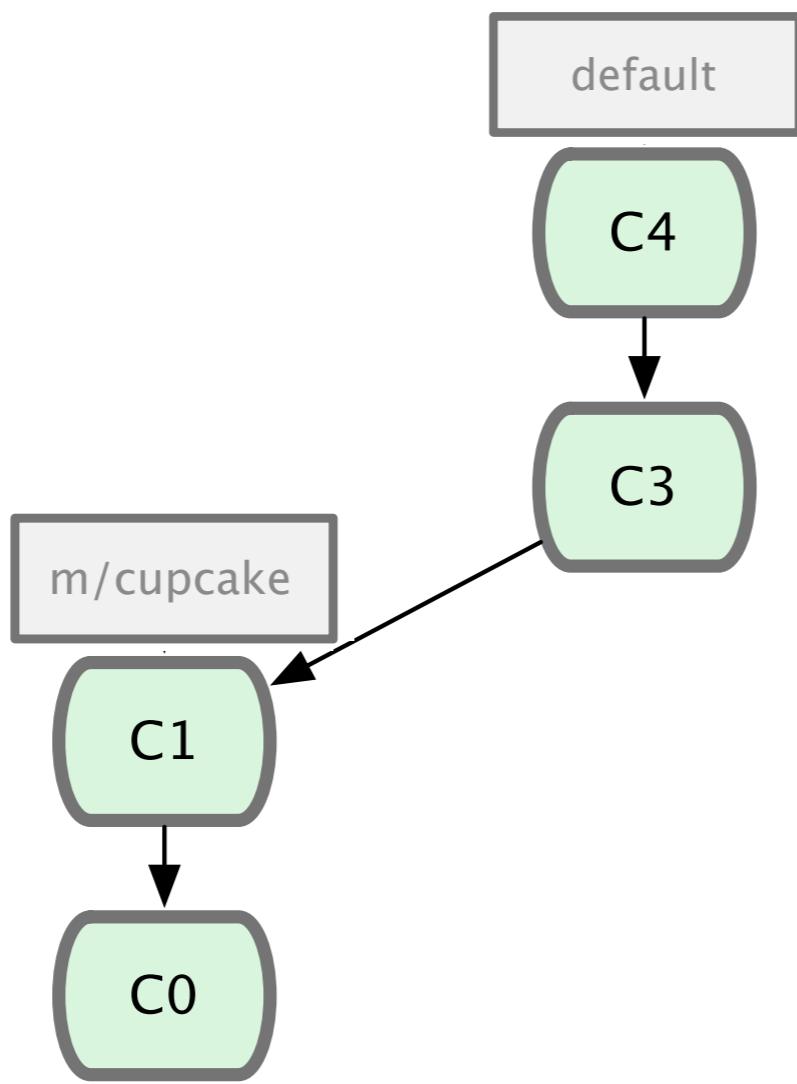
```
git log 3e45a ^573eb
```

```
git log 08cb1 ca374 ^c0bf9
```

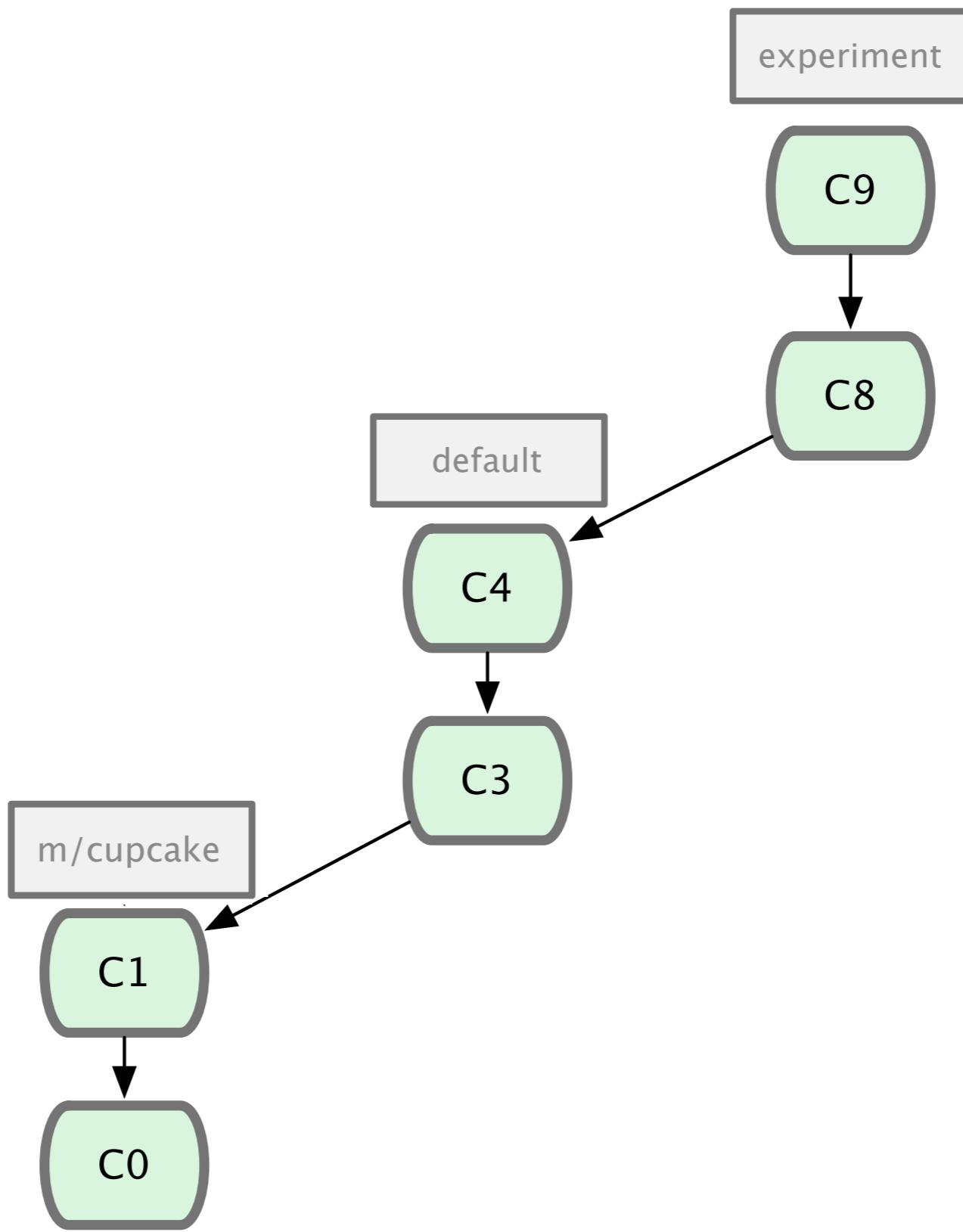
```
git log c5 c7 ^c9
```



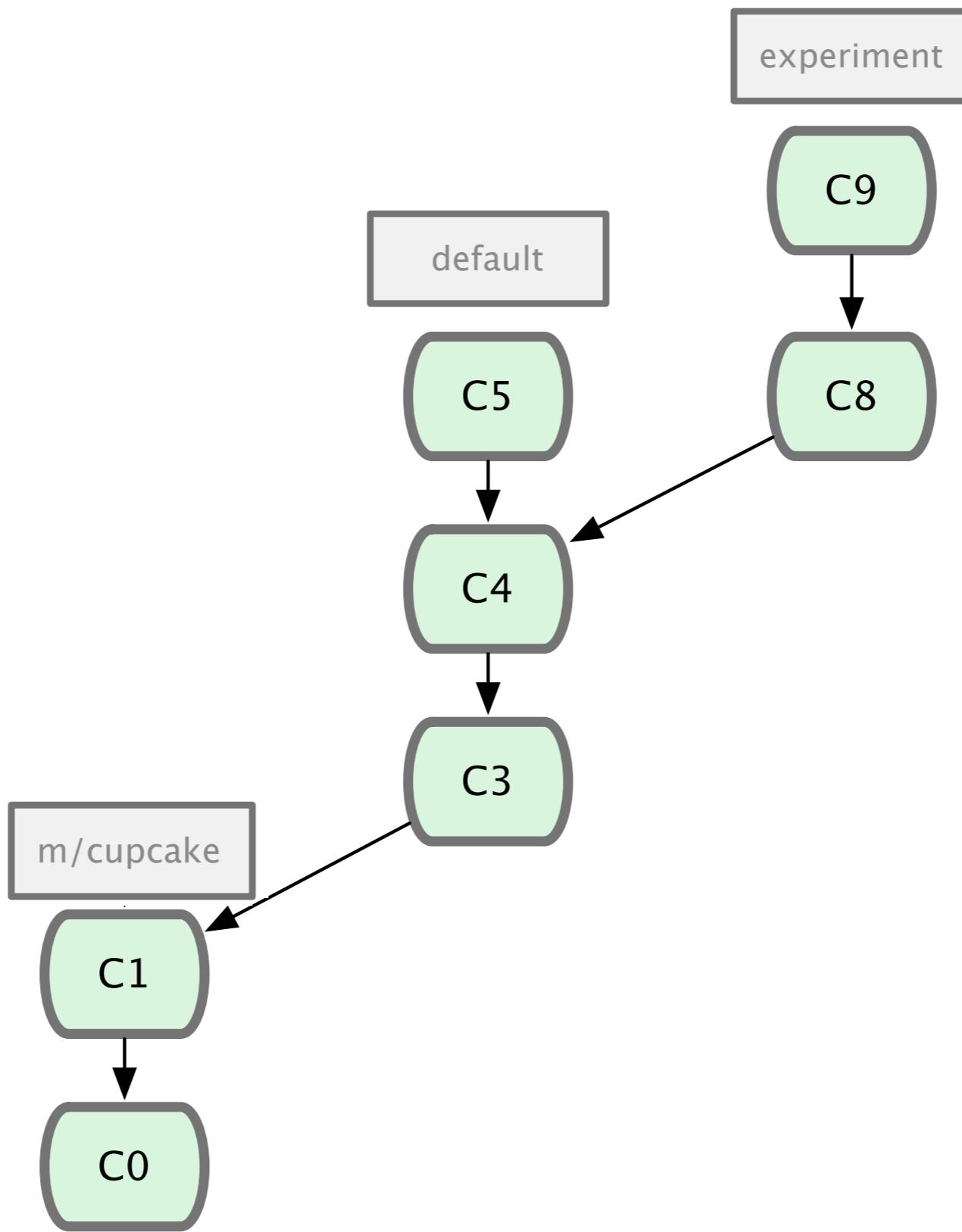
```
git log c5 c7 ^c9
```



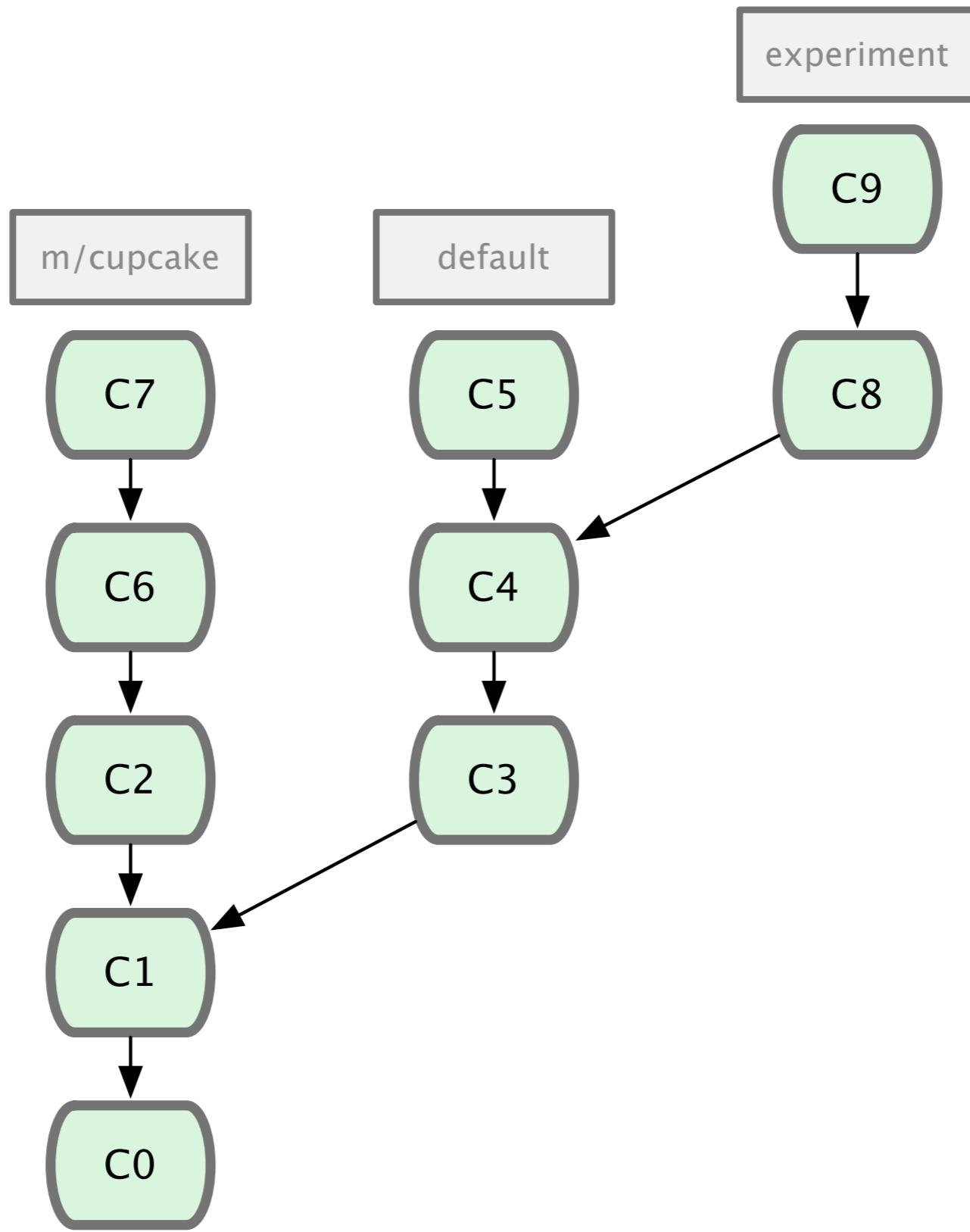
```
git log c5 c7 ^c9
```



```
git log c5 c7 ^c9
```

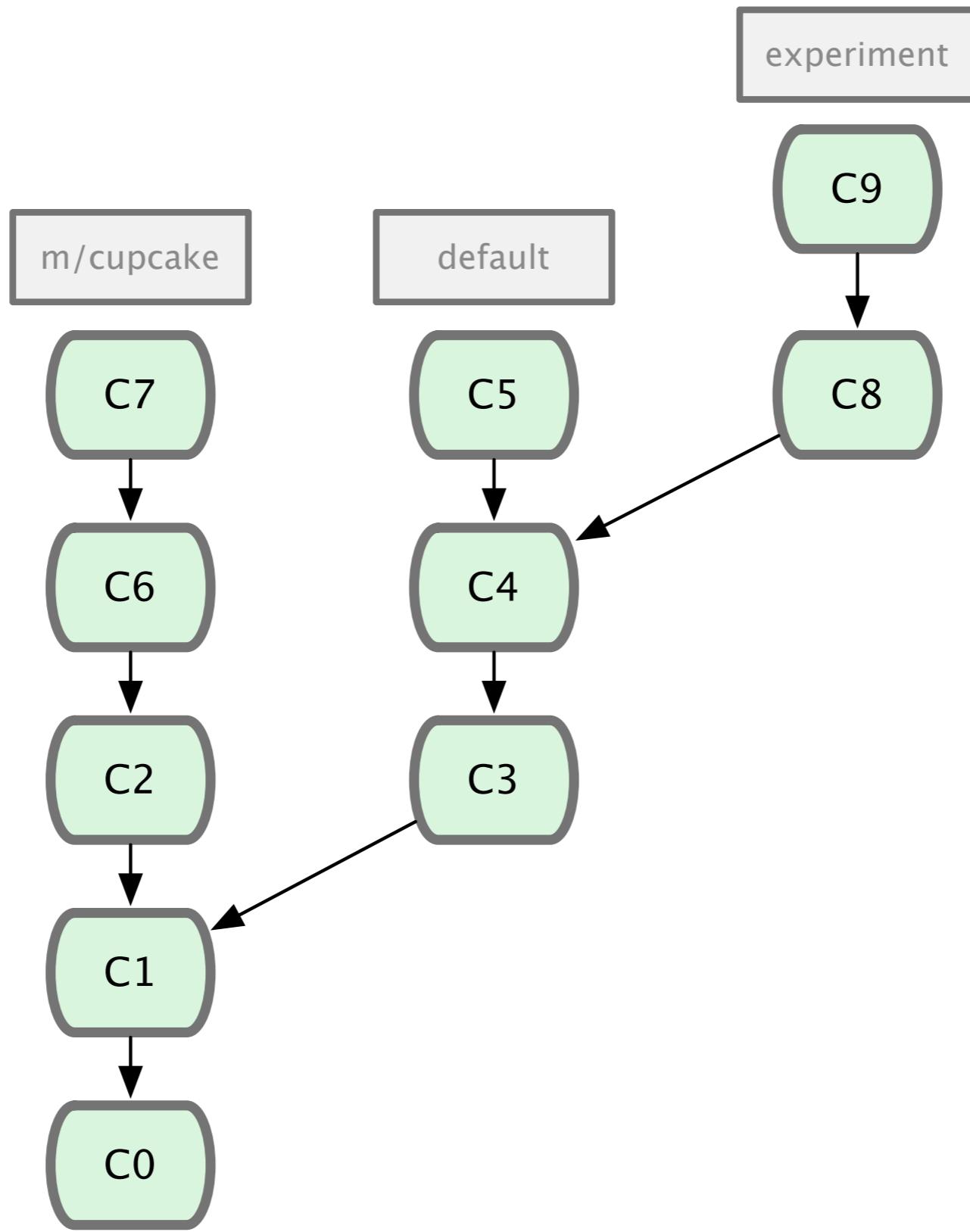


git log c5 c7 ^c9

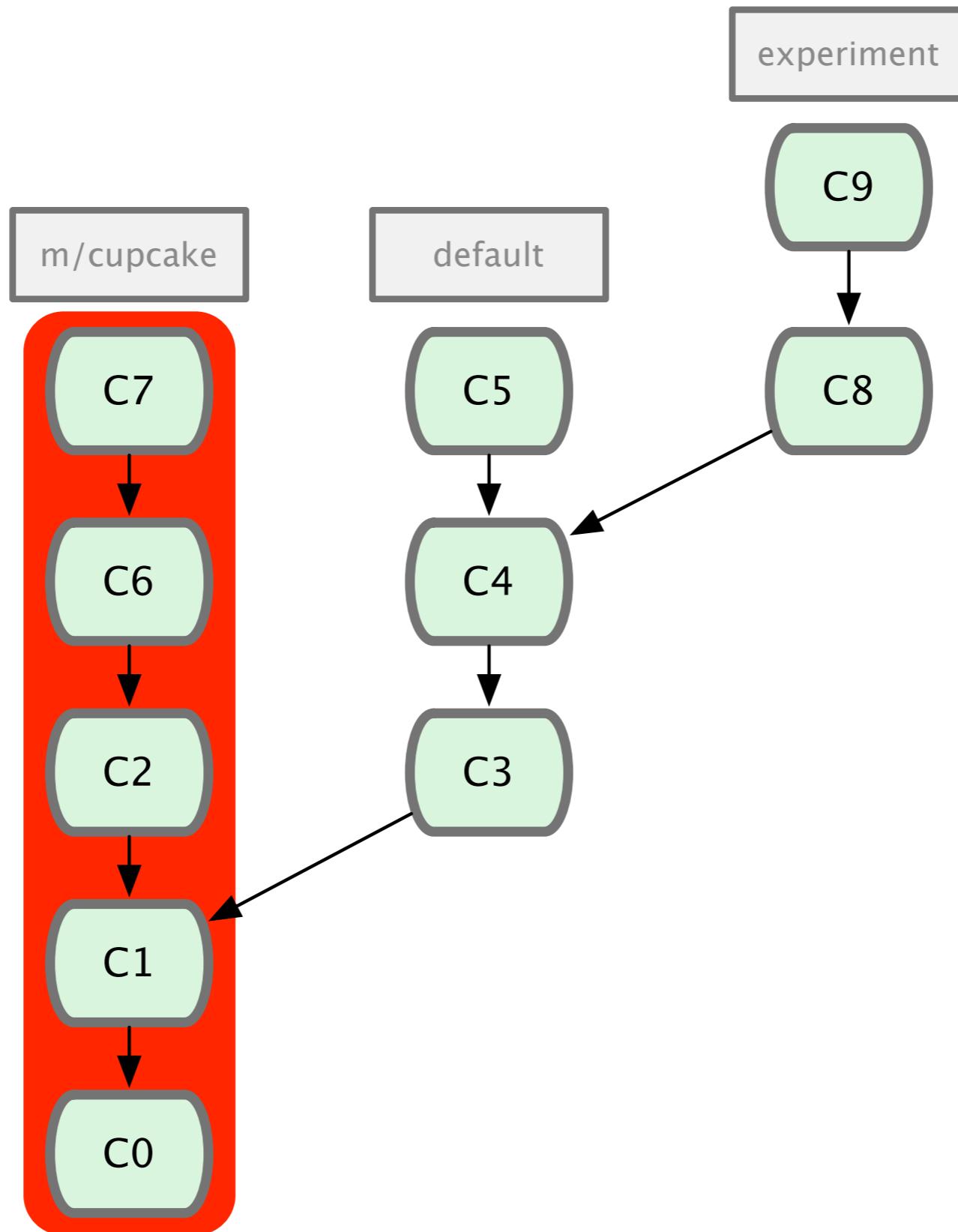


**which changes aren't in
experiment?**

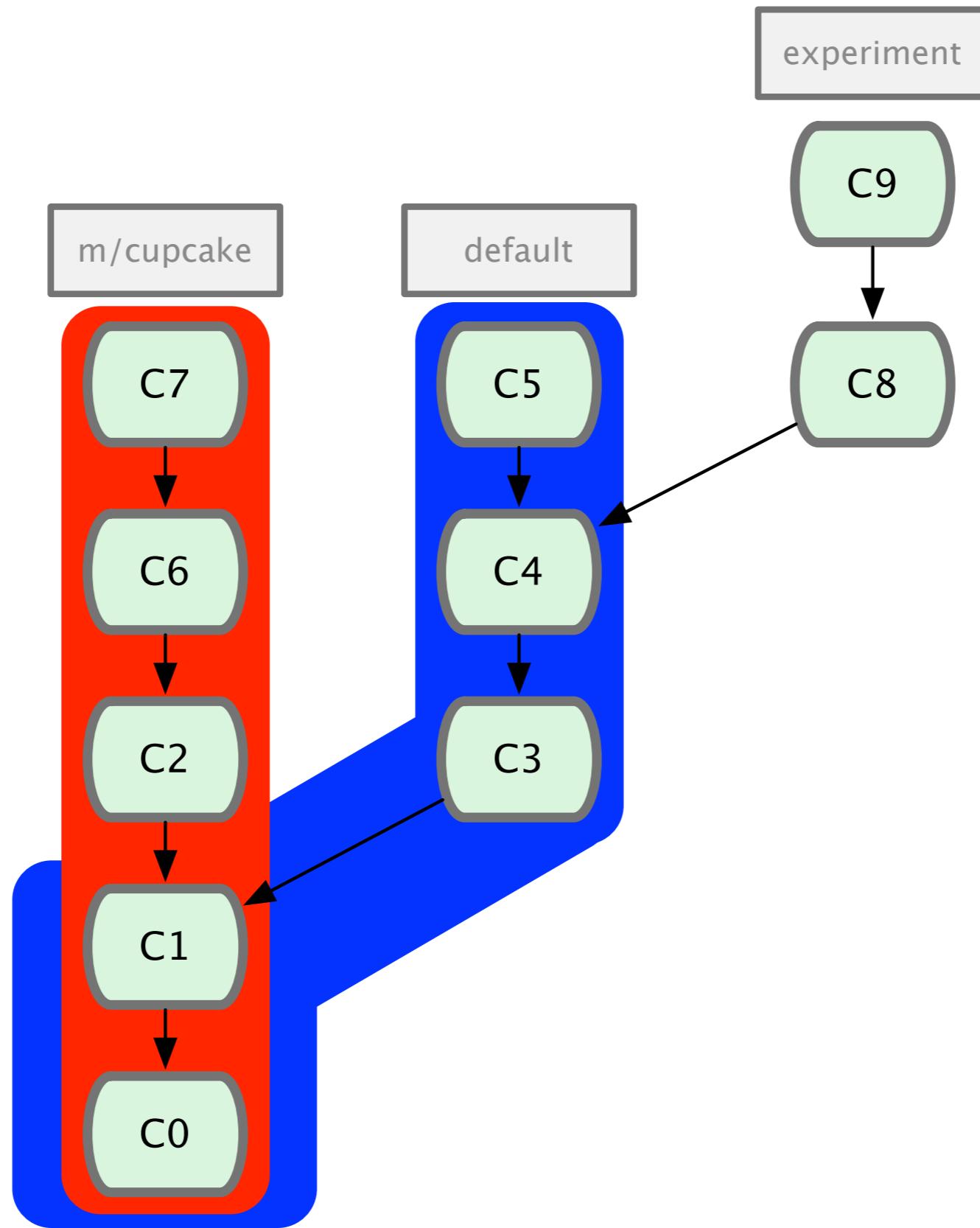
```
git log m/cupcake default ^experiment
```



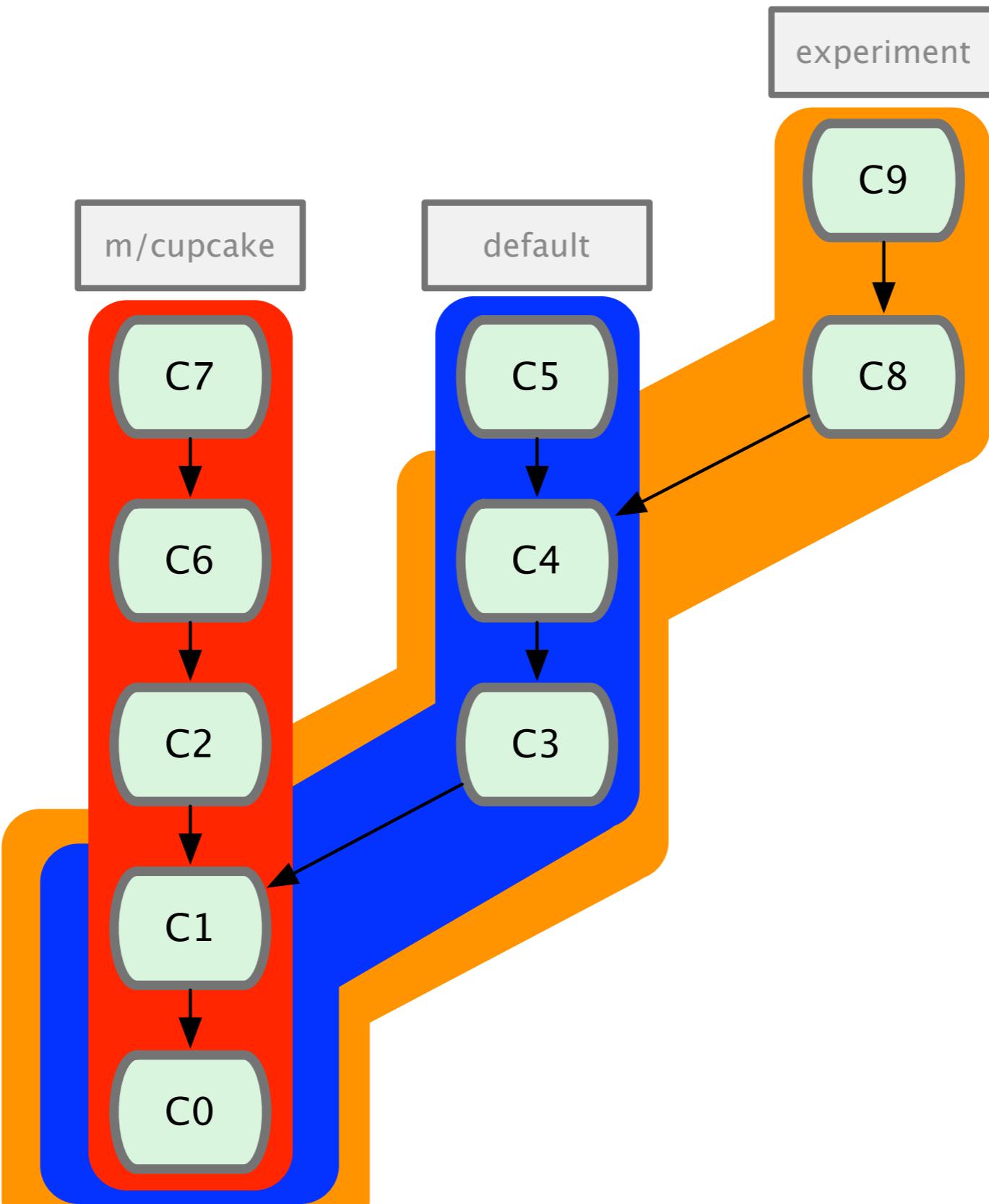
```
git log m/cupcake default ^experiment
```



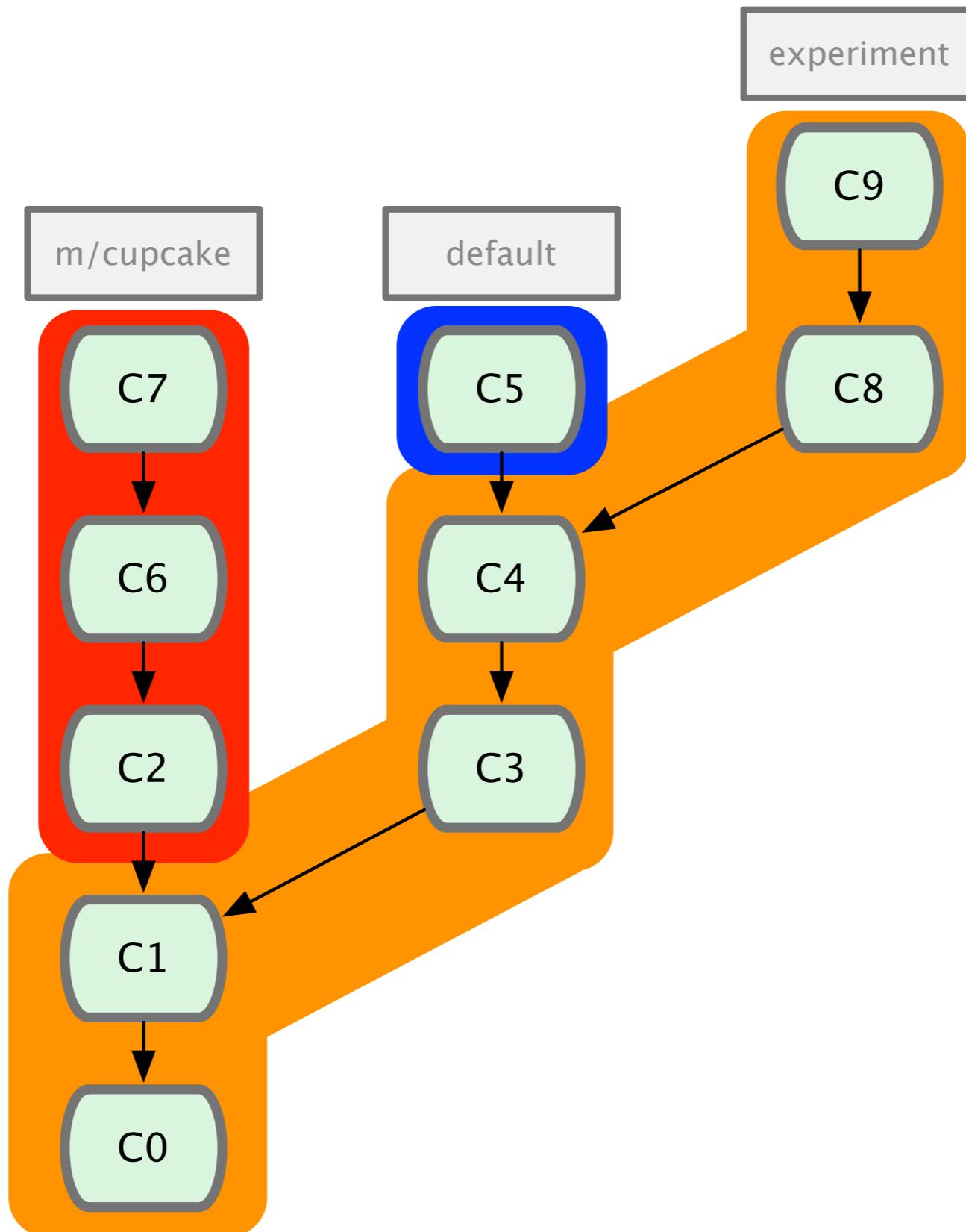
```
git log m/cupcake default ^experiment
```



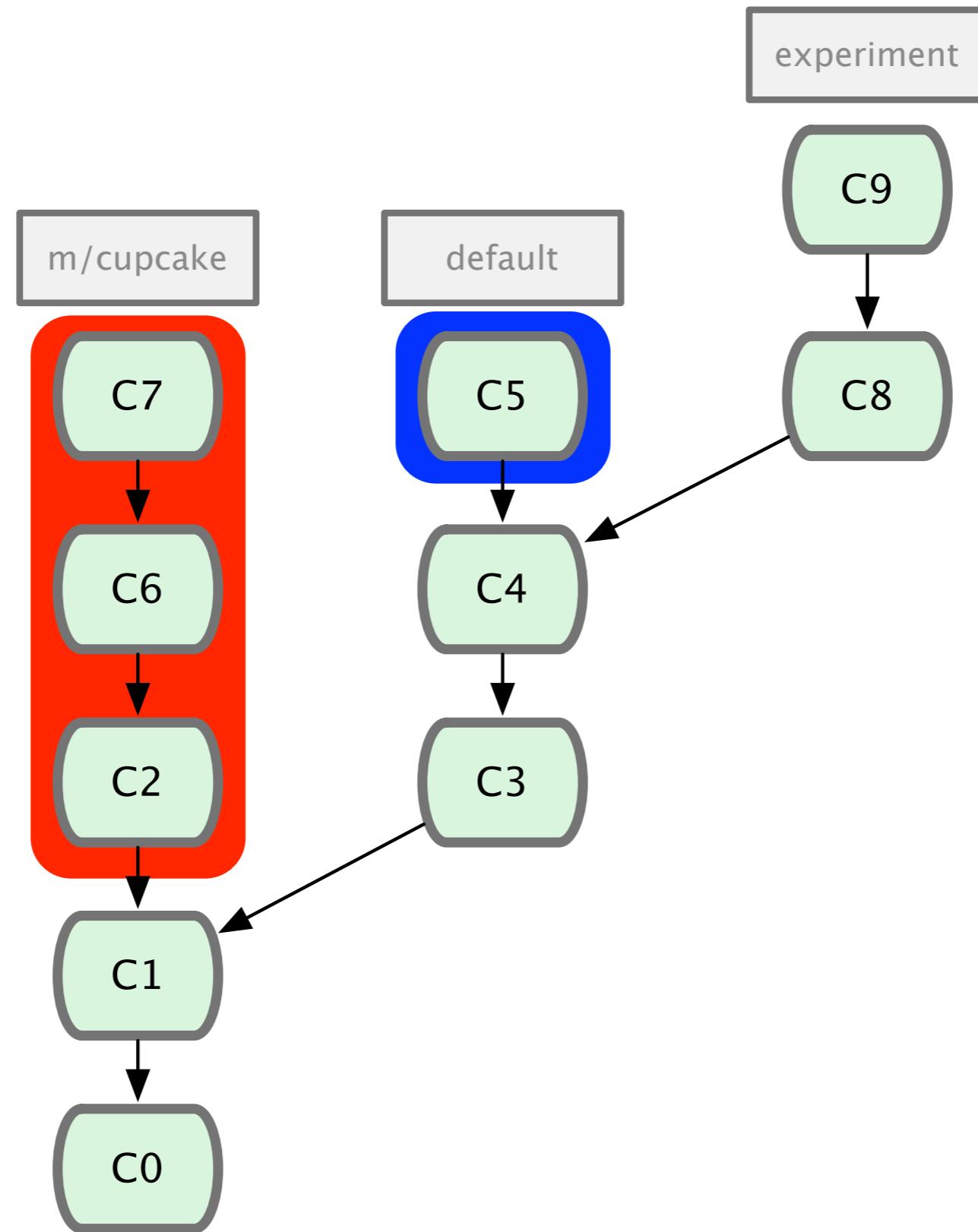
```
git log m/cupcake default ^experiment
```



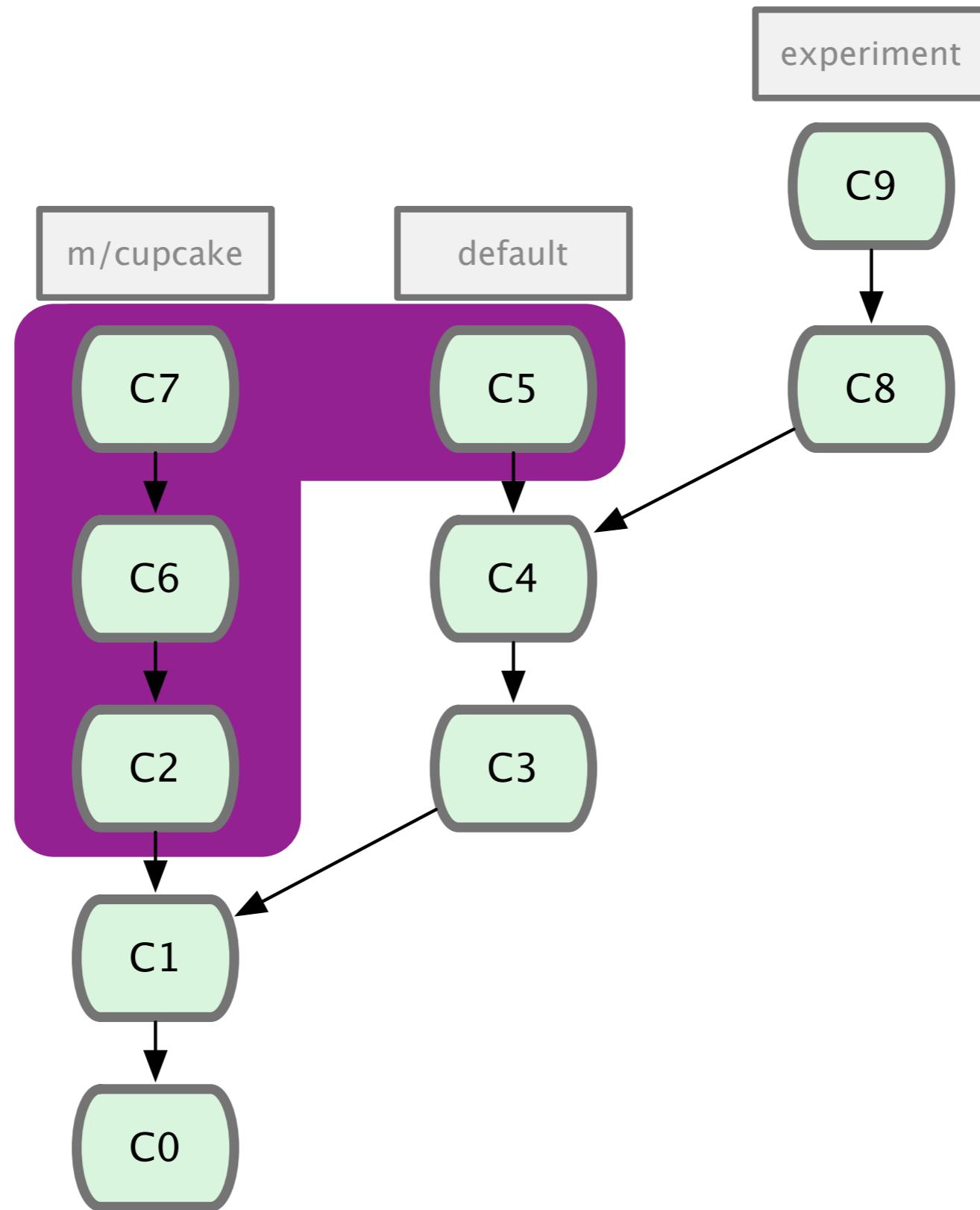
```
git log m/cupcake default ^experiment
```



```
git log m/cupcake default ^experiment
```



```
git log m/cupcake default ^experiment
```



Log Formatting

git log -P

```
commit 964f16d36dfccde844893cac5b347e7b3d44abbc
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 27 20:34:23 2008 -0700
```

commit the todo

```
diff --git a/TODO b/TODO
index da208f1..eaad748 100644
--- a/TODO
+++ b/TODO
@@ -3,3 +3,4 @@ Git Functions:
 * commit
 * rebase
 * merge
+* status
```

```
commit 8a5cbc430f1a9c3d00faaeffd07798508422908a
Author: Scott Chacon <schacon@gmail.com>
Date:   Wed Sep 3 16:10:57 2008 -0700
```

updated readme

```
diff --git a/README b/README
index 8ea1587..c526f88 100644
--- a/README
+++ b/README
@@ -1,9 +1,10 @@
 SimpleGit Ruby Library
 =====
```

```
-This library calls git commands and returns the output.
+This library calls git commands and returns the output. Sweet.
```

It is an example for the Git Peepcode book that I'm currently writing.

```
Author : Scott Chacon (schacon@gmail.com)
+      Orange Peel Chacon (opchacon@gmail.com)
      Magnus O. Chacon (mchacon@gmail.com)
```

```
commit a5f4a0daa9b13bb85283461ddaba6e589e34ccb5
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 13 10:51:45 2008 -0700
```

added cat-file

```
commit 964f16d36dfccde844893cac5b347e7b3d44abbc
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 27 20:34:23 2008 -0700

    commit the todo
```

```
diff --git a/TODO b/TODO
index da208f1..eaad748 100644
--- a/TODO
+++ b/TODO
@@ -3,3 +3,4 @@ Git Functions:
 * commit
 * rebase
 * merge
+* status
```

```
commit 8a5cbc430f1a9c3d00faaeffd07798508422908a
Author: Scott Chacon <schacon@gmail.com>
Date:   Wed Sep 3 16:10:57 2008 -0700
```

updated readme

```
diff --git a/README b/README
index 8ea1587..c526f88 100644
--- a/README
+++ b/README
@@ -1,9 +1,10 @@
 SimpleGit Ruby Library
 =====
```

```
-This library calls git commands and returns the output.
+This library calls git commands and returns the output. Sweet.
```

It is an example for the Git Peepcode book that I'm currently writing.

```
Author : Scott Chacon (schacon@gmail.com)
+      Orange Peel Chacon (opchacon@gmail.com)
      Magnus O. Chacon (mchacon@gmail.com)
```

```
commit a5f4a0daa9b13bb85283461ddaba6e589e34ccb5
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 13 10:51:45 2008 -0700
```

added cat-file

```
commit 964f16d36dfccde844893cac5b347e7b3d44abbc
Author: Magnus Chacon <mchacon@gmail.com>
Date: Sun Apr 27 20:34:23 2008 -0700
```

commit the todo

```
diff --git a/TODO b/TODO
index da208f1..eaad748 100644
--- a/TODO
+++ b/TODO
@@ -3,3 +3,4 @@ Git Functions:
 * commit
 * rebase
 * merge
+* status
```

```
commit 8a5cbc430f1a9c3d00faaeffd07798508422908a
Author: Scott Chacon <schacon@gmail.com>
Date: Wed Sep 3 16:10:57 2008 -0700
```

updated readme

```
diff --git a/README b/README
index 8ea1587..c526f88 100644
--- a/README
+++ b/README
@@ -1,9 +1,10 @@
 SimpleGit Ruby Library
 =====
```

```
-This library calls git commands and returns the output.
+This library calls git commands and returns the output. Sweet.
```

It is an example for the Git Peepcode book that I'm currently writing.

```
Author : Scott Chacon (schacon@gmail.com)
+ Orange Peel Chacon (opchacon@gmail.com)
Magnus O. Chacon (mchacon@gmail.com)
```

```
commit a5f4a0daa9b13bb85283461ddaba6e589e34ccb5
Author: Magnus Chacon <mchacon@gmail.com>
Date: Sun Apr 13 10:51:45 2008 -0700
```

added cat-file

```
commit 964f16d36dfccde844893cac5b347e7b3d44abbc
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 27 20:34:23 2008 -0700
```

commit the todo

```
diff --git a/TODO b/TODO
index da208f1..eaad748 100644
--- a/TODO
+++ b/TODO
@@ -3,3 +3,4 @@ Git Functions:
 * commit
 * rebase
 * merge
+* status
```

```
commit 8a5cbc430f1a9c3d00faaeffd07798508422908a
Author: Scott Chacon <schacon@gmail.com>
Date:   Wed Sep 3 16:10:57 2008 -0700
```

updated readme

```
diff --git a/README b/README
index 8ea1587..c526f88 100644
--- a/README
+++ b/README
@@ -1,9 +1,10 @@
 SimpleGit Ruby Library
 =====
```

```
-This library calls git commands and returns the output.
+This library calls git commands and returns the output. Sweet.
```

It is an example for the Git Peepcode book that I'm currently writing.

```
Author : Scott Chacon (schacon@gmail.com)
+      Orange Peel Chacon (opchacon@gmail.com)
      Magnus O. Chacon (mchacon@gmail.com)
```

```
commit a5f4a0daa9b13bb85283461ddaba6e589e34ccb5
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 13 10:51:45 2008 -0700
```

added cat-file

```
commit 964f16d36dfccde844893cac5b347e7b3d44abbc
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 27 20:34:23 2008 -0700
```

commit the todo

```
diff --git a/TODO b/TODO
index da208f1..eaad748 100644
--- a/TODO
+++ b/TODO
@@ -3,3 +3,4 @@ Git Functions:
 * commit
 * rebase
 * merge
+* status
```

```
commit 8a5cbc430f1a9c3d00faaeffd07798508422908a
Author: Scott Chacon <schacon@gmail.com>
Date:   Wed Sep 3 16:10:57 2008 -0700
```

updated readme

```
diff --git a/README b/README
index 8ea1587..c526f88 100644
--- a/README
+++ b/README
@@ -1,9 +1,10 @@
 SimpleGit Ruby Library
 =====
```

```
-This library calls git commands and returns the output.
+This library calls git commands and returns the output. Sweet.
```

It is an example for the Git Peepcode book that I'm currently writing.

```
Author : Scott Chacon (schacon@gmail.com)
+      Orange Peel Chacon (opchacon@gmail.com)
      Magnus O. Chacon (mchacon@gmail.com)
```

```
commit a5f4a0daa9b13bb85283461ddaba6e589e34ccb5
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 13 10:51:45 2008 -0700
```

added cat-file

git log -p

```
commit 964f16d36dfccde844893cac5b347e7b3d44abbc
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 27 20:34:23 2008 -0700
```

commit the todo

```
diff --git a/TODO b/TODO
index da208f1..eaad748 100644
--- a/TODO
+++ b/TODO
@@ -3,3 +3,4 @@ Git Functions:
 * commit
 * rebase
 * merge
+* status
```

```
commit 8a5cbc430f1a9c3d00faaeffd07798508422908a
Author: Scott Chacon <schacon@gmail.com>
Date:   Wed Sep 3 16:10:57 2008 -0700
```

updated readme

```
diff --git a/README b/README
index 8ea1587..c526f88 100644
--- a/README
+++ b/README
@@ -1,9 +1,10 @@
 SimpleGit Ruby Library
 =====
```

```
-This library calls git commands and returns the output.
+This library calls git commands and returns the output. Sweet.
```

It is an example for the Git Peepcode book that I'm currently writing.

```
Author : Scott Chacon (schacon@gmail.com)
+      Orange Peel Chacon (opchacon@gmail.com)
      Magnus O. Chacon (mchacon@gmail.com)
```

```
commit a5f4a0daa9b13bb85283461ddaba6e589e34ccb5
Author: Magnus Chacon <mchacon@gmail.com>
Date:   Sun Apr 13 10:51:45 2008 -0700
```

added cat-file

git log --pretty

```
$>git log --pretty=oneline
```

```
$>git log --pretty=oneline
```

3e625e2139a71bfbaa85b055c139bceafab7e2c0	updated rakefile
4fefbaafd51ff3e79b84e9ee1a0dac2848e21a98	commit the todo
c4f8fd68ae3897d1d10456ed2b93d87cdc53d6e8	updated README
a5f4a0daa9b13bb85283461ddaba6e589e34ccb5	added cat-file
310154e3c7db47d8bac935c2c43aee6afac11aae	updated README formatting
f7f3f6dd8fd3fa40f052427c32785a0fa01aaa5f	changed my name a bit
710f0f8d2cdf5af87033b9ec08859a505f9a6af5	added ls-files
c110d7ff8cfb86fd5cce9a8aee462678dbb4ef9b	made the ls-tree function
ce9b0d5551762048735dd67917046b44176317e0	limiting log to 30
e22e01e39c39521edc8ccd3aa5df228209e7ad2d	-added todo options
2c0d4d7e3d36a8cd5daleab3e0c1e83faa3bfd9f	added limit to log function
47c668a9e8f93f65ff9019c587f3b2204bd6c98d	fixed conflict
ca28a28e64192af4527ea63665dceaa9bdb6a4b	new version 0.1.2
f30ba8480d9aba88580696651e80d9aab8e15a2	new version 0.2.0
1edee6b1d61823a2de3b09c160d7080b8d1b3a40	added a new function
4b0780c5b7a4ae1217f4da742a172e7d787ca3eb	rakefile and todo file add
0d57a4eb03d35e1e1041857d1cb01199eeb23cc4	Revert "rakefile and todo
f65a297eb0ace63e5b302e849064f37ded909c0b	rakefile and todo file add
c035c611b559537d725d875cfa10c05c17d48e4f	Merge branch 'newfunc'
d6fad7dd6f98481cab874b192aaa79d03490d4a0	added more description
1a8c32ea9725886aaadc35dbf4d68468d3af0abe	added lmtree function
40e5c10f67fa01c01da5687d08bd3bc21bb6a44b	added email to readme
cf25cc3bfb0ece7dc3609b8dc0cc4a1e19ffbcd4	committing all changes
0c8a9ec46029a4e92a428cb98c9693f09f69a3ff	changed the verison number

```
$>git log --pretty=format:"%h %an %ar - %s"__
```

```
$>git log --pretty=format:"%h %an %ar - %s"
3e625e2 Magnus Chacon 47 minutes ago - updated rakefile
4fefbaa Magnus Chacon 56 minutes ago - commit the todo
c4f8fd6 Magnus Chacon 60 minutes ago - updated README
a5f4a0d Magnus Chacon 2 weeks ago - added cat-file
310154e Scott Chacon 2 weeks ago - updated README formatting and a
f7f3f6d Magnus Chacon 2 weeks ago - changed my name a bit
710f0f8 Magnus Chacon 2 weeks ago - added ls-files
c110d7f Scott Chacon 3 weeks ago - made the ls-tree function recur
ce9b0d5 Scott Chacon 3 weeks ago - limiting log to 30
e22e01e Scott Chacon 3 weeks ago - -added todo options
2c0d4d7 Scott Chacon 3 weeks ago - added limit to log function
47c668a Scott Chacon 3 weeks ago - fixed conflict
ca28a28 Scott Chacon 3 weeks ago - new version 0.1.2
f30ba84 Scott Chacon 3 weeks ago - new version 0.2.0
1edee6b Scott Chacon 3 weeks ago - added a new function
4b0780c Scott Chacon 3 weeks ago - rakefile and todo file added
0d57a4e Scott Chacon 3 weeks ago - Revert "rakefile and todo file
f65a297 Scott Chacon 3 weeks ago - rakefile and todo file added
c035c61 Scott Chacon 3 weeks ago - Merge branch 'newfunc'
d6fad7d Scott Chacon 3 weeks ago - added more description
1a8c32e Scott Chacon 3 weeks ago - added lstree function
40e5c10 Scott Chacon 3 weeks ago - added email to readme
cf25cc3 Scott Chacon 6 weeks ago - committing all changes
0c8a9ec Scott Chacon 6 weeks ago - changed the verison number
0576fac Scott Chacon 6 weeks ago - my second commit which is bett
```

```
$>git log --pretty=format:"%h %an %ar - %s"
3e625e2 Magnus Chacon 47 minutes ago - updated rakefile
4fefbaa Magnus Chacon 56 minutes ago - commit the todo
c4f8fd6 Magnus Chacon 60 minutes ago - updated README
a5f4a0d Magnus Chacon 2 weeks ago - added cat-file
310154e Scott Chacon 2 weeks ago - updated README formatting and a
f7f3f6d Magnus Chacon 2 weeks ago - changed my name a bit
710f0f8 Magnus Chacon 2 weeks ago - added ls-files
c110d7f Scott Chacon 3 weeks ago - made the ls-tree function recur
ce9b0d5 Scott Chacon 3 weeks ago - limiting log to 30
e22e01e Scott Chacon 3 weeks ago - -added todo options
2c0d4d7 Scott Chacon 3 weeks ago - added limit to log function
47c668a Scott Chacon 3 weeks ago - fixed conflict
ca28a28 Scott Chacon 3 weeks ago - new version 0.1.2
f30ba84 Scott Chacon 3 weeks ago - new version 0.2.0
1edee6b Scott Chacon 3 weeks ago - added a new function
4b0780c Scott Chacon 3 weeks ago - rakefile and todo file added
0d57a4e Scott Chacon 3 weeks ago - Revert "rakefile and todo file
f65a297 Scott Chacon 3 weeks ago - rakefile and todo file added
c035c61 Scott Chacon 3 weeks ago - Merge branch 'newfunc'
d6fad7d Scott Chacon 3 weeks ago - added more description
1a8c32e Scott Chacon 3 weeks ago - added lstree function
40e5c10 Scott Chacon 3 weeks ago - added email to readme
cf25cc3 Scott Chacon 6 weeks ago - committing all changes
0c8a9ec Scott Chacon 6 weeks ago - changed the verison number
0576fac Scott Chacon 6 weeks ago - my second commit which is bett
```

```
$>git log --pretty=format:"%h %an %ar - %s"
3e625e2 Magnus Chacon 47 minutes ago - updated rakefile
4fefbaa Magnus Chacon 56 minutes ago - commit the todo
c4f8fd6 Magnus Chacon 60 minutes ago - updated README
a5f4a0d Magnus Chacon 2 weeks ago - added cat-file
310154e Scott Chacon 2 weeks ago - updated README formatting and a
f7f3f6d Magnus Chacon 2 weeks ago - changed my name a bit
710f0f8 Magnus Chacon 2 weeks ago - added ls-files
c110d7f Scott Chacon 3 weeks ago - made the ls-tree function recur
ce9b0d5 Scott Chacon 3 weeks ago - limiting log to 30
e22e01e Scott Chacon 3 weeks ago - -added todo options
2c0d4d7 Scott Chacon 3 weeks ago - added limit to log function
47c668a Scott Chacon 3 weeks ago - fixed conflict
ca28a28 Scott Chacon 3 weeks ago - new version 0.1.2
f30ba84 Scott Chacon 3 weeks ago - new version 0.2.0
1edee6b Scott Chacon 3 weeks ago - added a new function
4b0780c Scott Chacon 3 weeks ago - rakefile and todo file added
0d57a4e Scott Chacon 3 weeks ago - Revert "rakefile and todo file
f65a297 Scott Chacon 3 weeks ago - rakefile and todo file added
c035c61 Scott Chacon 3 weeks ago - Merge branch 'newfunc'
d6fad7d Scott Chacon 3 weeks ago - added more description
1a8c32e Scott Chacon 3 weeks ago - added lstree function
40e5c10 Scott Chacon 3 weeks ago - added email to readme
cf25cc3 Scott Chacon 6 weeks ago - committing all changes
0c8a9ec Scott Chacon 6 weeks ago - changed the verison number
0576fac Scott Chacon 6 weeks ago - my second commit which is bett
```

git log --left-right

```
$ git log --pretty=oneline --left-right 5f7f801...77976b
```

```
$ git log --pretty=oneline --left-right 5f7f801...77976b
<5f7f801227868c7abcce7e58dee3eff855011955 convert file browser to html
>77976b24ff839c59c3b20d80cb28351ccb5e59a8 remove cruft
>09b76d2966e2370a78ed37a31c2f7c23d08609c3 git-log
>7000b24511618a21d40b39ee213d397e1d29497d like i said
>c2a6adfdc18c0d95dbed6ea62ac9c9a912d18123 create this on deploy
>6ba3609953d5c46a76ca1d0d3d83018be61454e6 i am too meta for you
<3dff6074fe205e36fae219f277ef87aab097e236 Merge branch 'master' into I
<1cdc8437fa6c621d96c4dfa5f6370c8fdb9cbc3d Merge branch 'master' into I
<20125a6d37d5c1614ffelde94ca064095968e7f0 delete default images
```

```
$ git log --pretty=oneline --left-right 5f7f801...77976b
<5f7f801227868c7abcce7e58dee3eff855011955 convert file browser to html
>77976b24ff839c59c3b20d80cb28351ccb5e59a8 remove cruft
>09b76d2966e2370a78ed37a31c2f7c23d08609c3 git-log
>7000b24511618a21d40b39ee213d397e1d29497d like i said
>c2a6adfdc18c0d95dbed6ea62ac9c9a912d18123 create this on deploy
>6ba3609953d5c46a76ca1d0d3d83018be61454e6 i am too meta for you
<3dff6074fe205e36fae219f277ef87aab097e236 Merge branch 'master' into I
<1cdc8437fa6c621d96c4dfa5f6370c8fdb9cbc3d Merge branch 'master' into I
<20125a6d37d5c1614ffelde94ca064095968e7f0 delete default images
```

```
$ git log --pretty=oneline --left-right 5f7f801...77976b
<5f7f801227868c7abcce7e58dee3eff855011955 convert file browser to html
>77976b24ff839c59c3b20d80cb28351ccb5e59a8 remove cruft
>09b76d2966e2370a78ed37a31c2f7c23d08609c3 git-log
>7000b24511618a21d40b39ee213d397e1d29497d like i said
>c2a6adfdc18c0d95dbed6ea62ac9c9a912d18123 create this on deploy
>6ba3609953d5c46a76ca1d0d3d83018be61454e6 i am too meta for you
<3dff6074fe205e36fae219f277ef87aab097e236 Merge branch 'master' into I
<1cdc8437fa6c621d96c4dfa5f6370c8fdb9cbc3d Merge branch 'master' into I
<20125a6d37d5c1614ffelde94ca064095968e7f0 delete default images
```

`git log --graph`

```
git log --pretty=oneline --graph
```

```
* 20d4111 - fix copy strategy for production
* 86ef476 - confirm daemon servering is on for public imports
* 990bc8b - Merge branch 'friendly_delete'
  \
  | * ed555b7 - better deletion flow
  | * eec7491 - verify private forks aren't deleted
  | * d65fc92 - simple fork deletion logic
  | * e9370f4 - one more test for the gipper
  | * 1d7c276 - remove busted test, add deleted template test
  | * c0bf99f - let forks know their private network was deleted
  * | ef2a5e2 - out of sight, out of mind
  * | d352768 - guess we can turn this off now
  * | d0f25c0 - use default timeout
  * | 2e6eeae - convert executes to AR:B
  * | e6b8628 - Merge branch 'mirrors'
    \
    | * | ca3744f - couple mirror bug fixes
    | * | 56b8868 - mirroring
    * | | 08cb112 - bang. ya dead.
    / /
    * | c153b21 - let AR:B figure it out
    * | 19e66a2 - raw private gists are ssl'd
    * | fdda4ac - Merge branch 'master' of git@github.com:defunkt/github
      \
      | * | 0e9cf1 - x add slice39 to the deploy role
      | * | 82703eb - Revert "update newrelic plugin"
      | * | 77faff1 - turn on tracer
      | * | 3807239 - update newrelic plugin
      * | | 0aaehh2 - sexy up inbox messages
```

gitk

gitk: github

fix copy strategy for production
confirm daemon servering is on for public imports
Merge branch 'friendly_delete'
better deletion flow
verify private forks aren't deleted
simple fork deletion logic
one more test for the gipper
remove busted test, add deleted template test
let forks know their private network was deleted
out of sight, out of mind
guess we can turn this off now
use default timeout
convert executes to AR:B
Merge branch 'mirrors'
couple mirror bug fixes
mirroring
bang. ya dead.
let AR:B figure it out
raw private gists are ssl'd
Merge branch 'master' of git@github.com:defunkt/github
x add slice39 to the deploy role
Report "undate no url fix plugin"

PJ Hyett <pjhyett@gmail.com> 2008-09-30 16:49:34
PJ Hyett <pjhyett@gmail.com> 2008-09-29 18:17:19
PJ Hyett <pjhyett@gmail.com> 2008-09-29 16:07:04
PJ Hyett <pjhyett@gmail.com> 2008-09-29 16:05:17
PJ Hyett <pjhyett@gmail.com> 2008-09-29 15:31:02
PJ Hyett <pjhyett@gmail.com> 2008-09-29 15:13:51
PJ Hyett <pjhyett@gmail.com> 2008-09-22 17:03:25
PJ Hyett <pjhyett@gmail.com> 2008-09-22 16:52:40
PJ Hyett <pjhyett@gmail.com> 2008-09-19 18:31:50
PJ Hyett <pjhyett@gmail.com> 2008-09-29 12:46:18
PJ Hyett <pjhyett@gmail.com> 2008-09-29 12:09:16
PJ Hyett <pjhyett@gmail.com> 2008-09-29 11:57:49
PJ Hyett <pjhyett@gmail.com> 2008-09-26 17:46:50
PJ Hyett <pjhyett@gmail.com> 2008-09-26 15:32:34
PJ Hyett <pjhyett@gmail.com> 2008-09-26 15:32:09
PJ Hyett <pjhyett@gmail.com> 2008-09-26 15:17:37
Chris Wanstrath <chris@ozmm.org> 2008-09-26 14:46:15
PJ Hyett <pjhyett@gmail.com> 2008-09-24 18:29:54
Chris Wanstrath <chris@ozmm.org> 2008-09-24 14:31:31
Chris Wanstrath <chris@ozmm.org> 2008-09-23 18:48:04
PJ Hyett <pjhyett@gmail.com> 2008-09-23 17:35:29
PJ Hyett <pjhyett@gmail.com> 2008-09-23 17:28:58

SHA1 ID: 20d4113a586b0ed05f530e6f7f4b895540da0f0

Find next prev commit containing: Exact All fields

Search Patch Tree

Diff Old version New version Lines of context: 3 Ignore space change

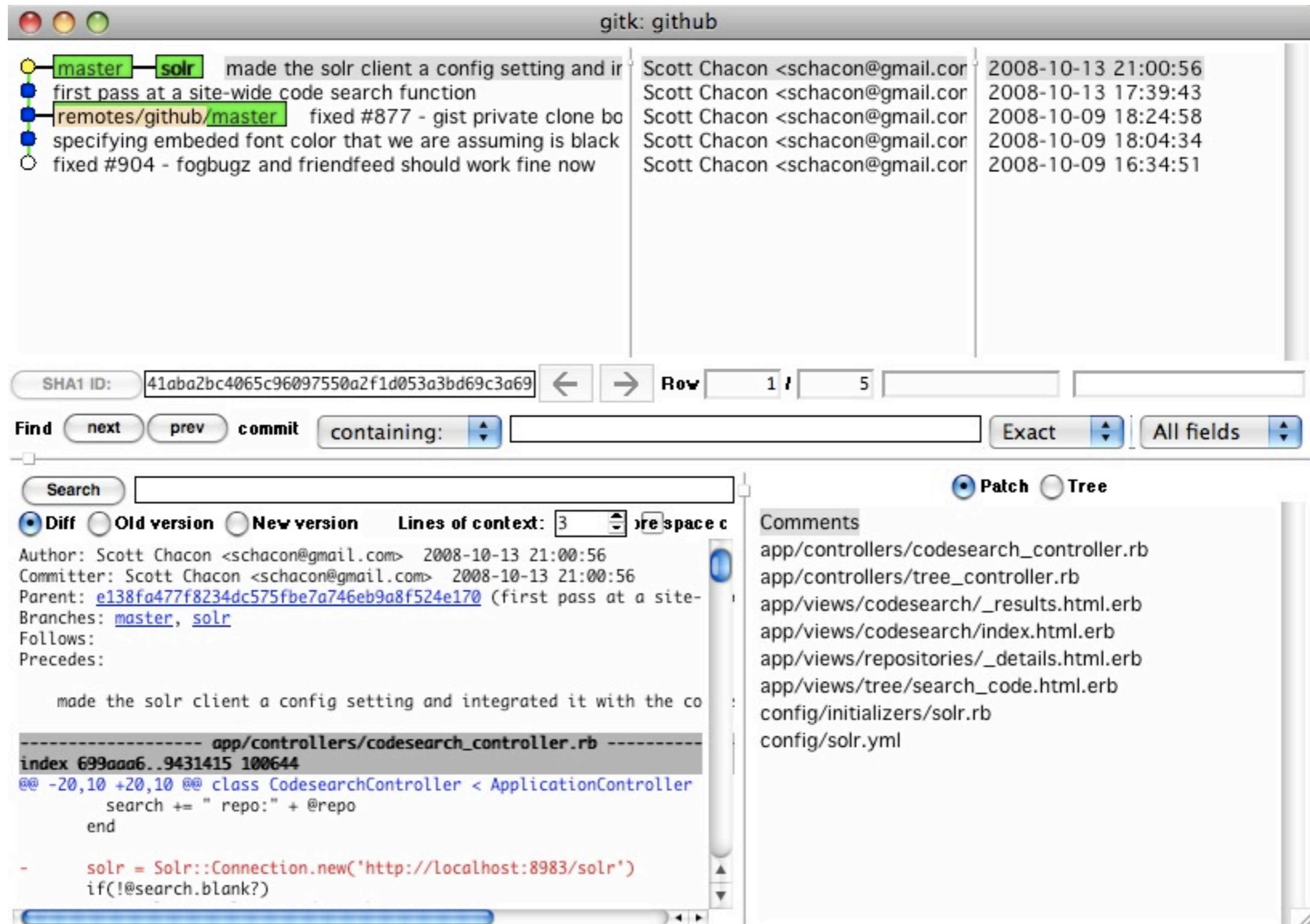
Author: PJ Hyett <pjhyett@gmail.com> 2008-09-30 16:49:34
Committer: PJ Hyett <pjhyett@gmail.com> 2008-09-30 16:49:34
Parent: [86ef47642bf2a1c67820de999e15437601e872bb](#) (confirm daemon servering is on for public
Branches: master, remotes/github/master, solr, training, wiki2
Follows:
Precedes:

fix copy strategy for production

----- app/models/repository.rb -----
index 09b8659..574ab6b 100644
@@ -82,11 +82,7 @@ class Repository < ActiveRecord::Base
end

Comments
app/models/repository.rb
jobs/repository-delete
test/unit/repository_delete_job_test.rb

gitk 6d13f..41aba



Merged Branches

git branch --merged

git branch --no-merged

Branch Diffs

git diff branch

```
diff --git a/Rakefile b/Rakefile
index eb54561..2d2c777 100644
--- a/Rakefile
+++ b/Rakefile
@@ -17,10 +17,6 @@ Rake::GemPackageTask.new(spec) do |pkg|
    pkg.need_tar = true
end

-task :default => "pkg/#{spec.name}-#{spec.version}.gem" do
-  puts "generated latest version"
-end

-
desc "Regenerate Documentation"
task :doc do |t|
  system('rdoc lib/ README --main README --inline-source')
diff --git a/TODO b/TODO
deleted file mode 100644
index bbe7708..0000000
--- a/TODO
+++ /dev/null
@@ -1,7 +0,0 @@
-Git Functions:
-* add
```

`git diff --stat`

```
$ git diff --stat branch1
Rakefile           |   4 -----
TODO              |   7 -----
lib/simple_git.rb |  48 -----
lib/simplegit.rb  |  52 ++++++-----+
4 files changed, 52 insertions(+), 59 deletions(-)
```

```
git diff --name-only
```

```
$ git diff --name-only branch  
Rakefile  
TODO  
lib/simple_git.rb  
lib/simplegit.rb
```

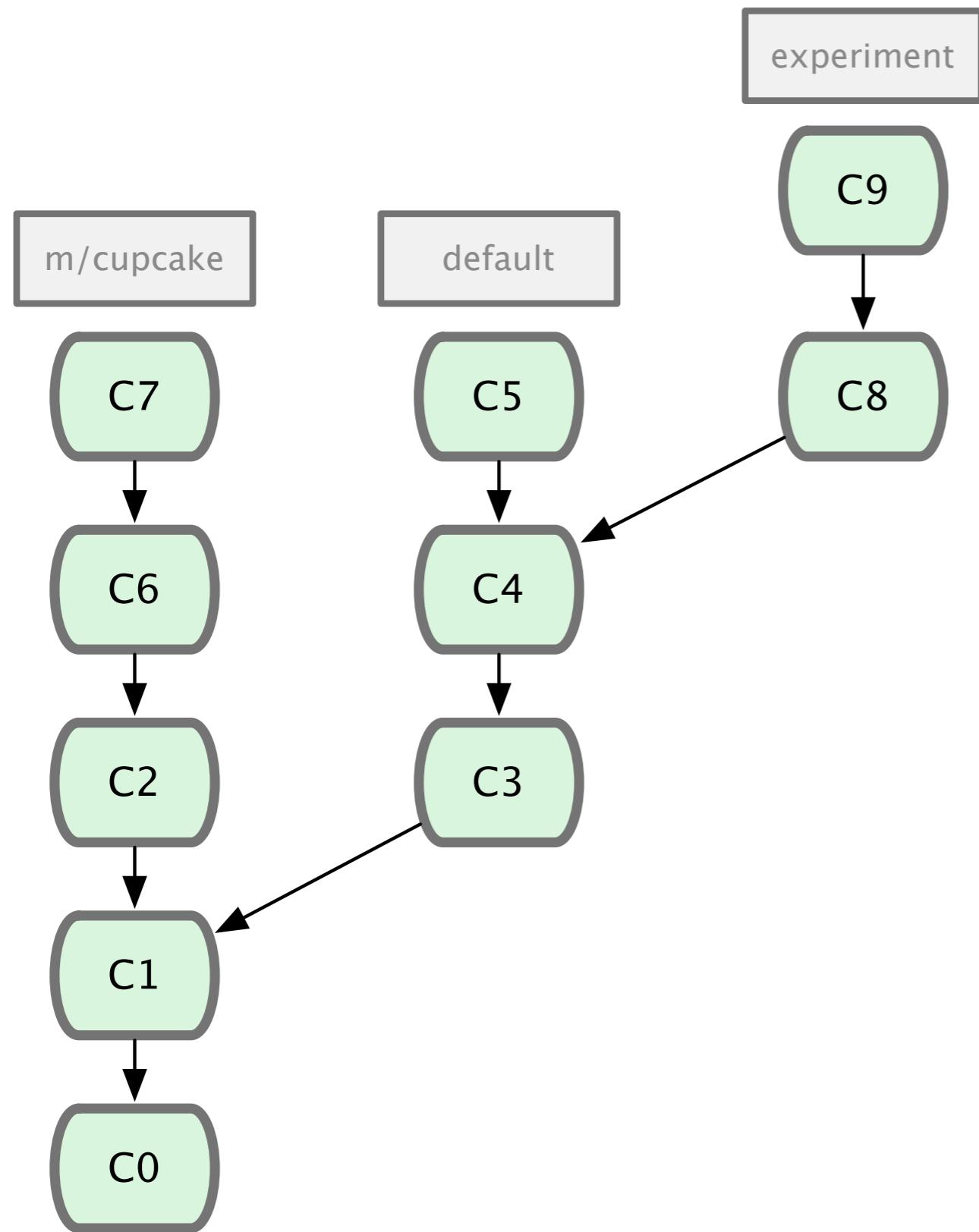
git diff branch

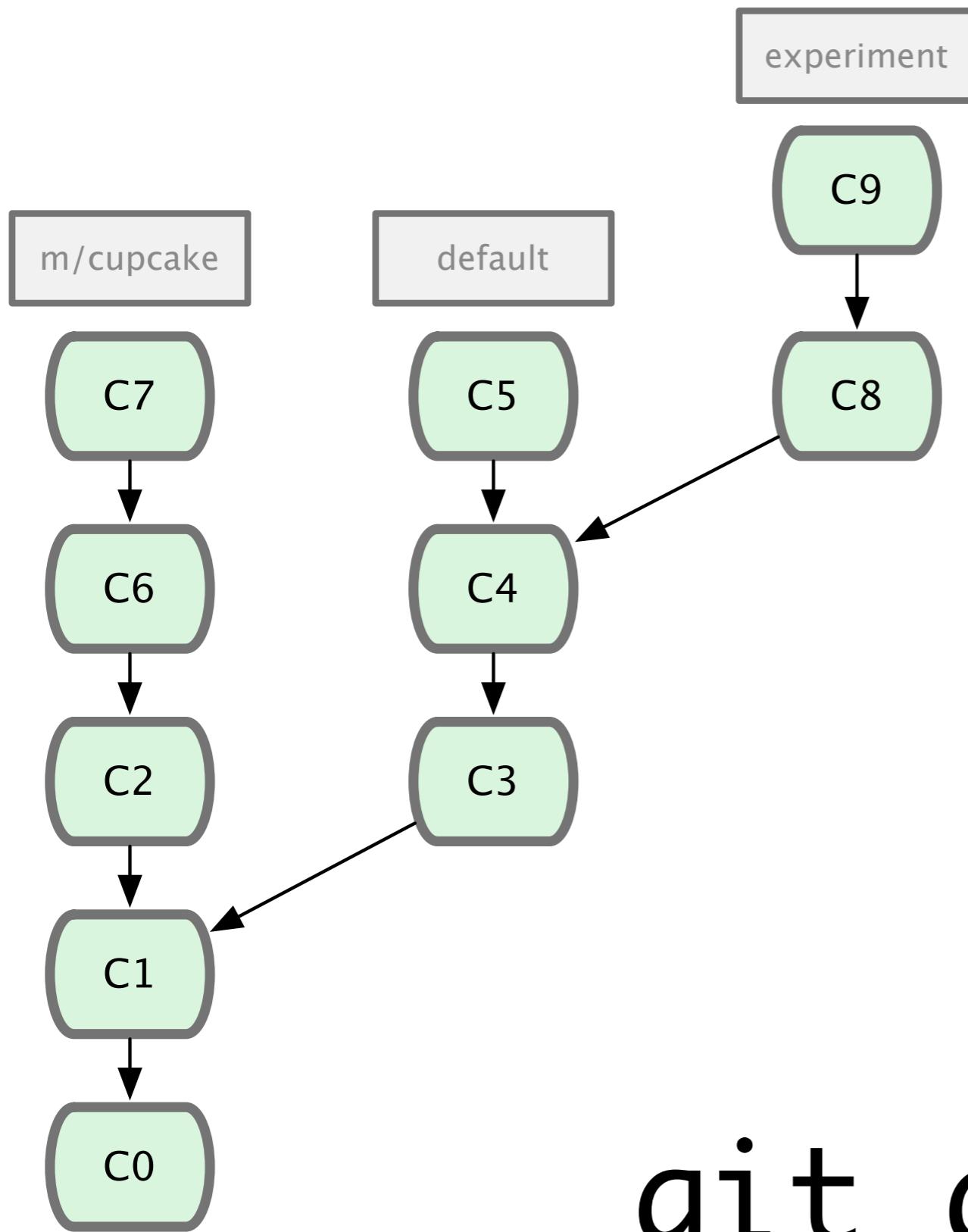
`git diff branch HEAD`

```
git diff branch1 branch2
```

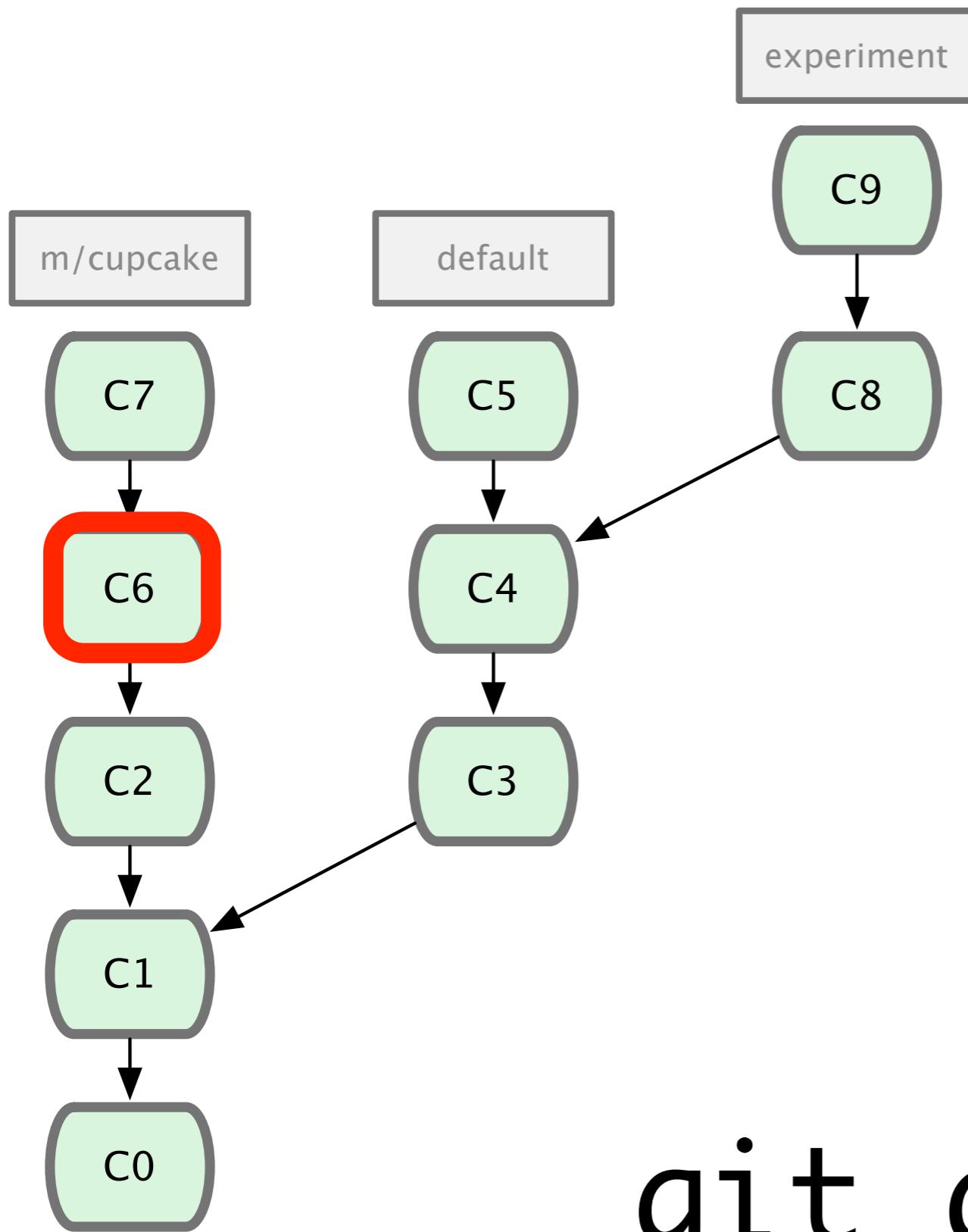
Revision Diffs

```
git diff revA revB
```

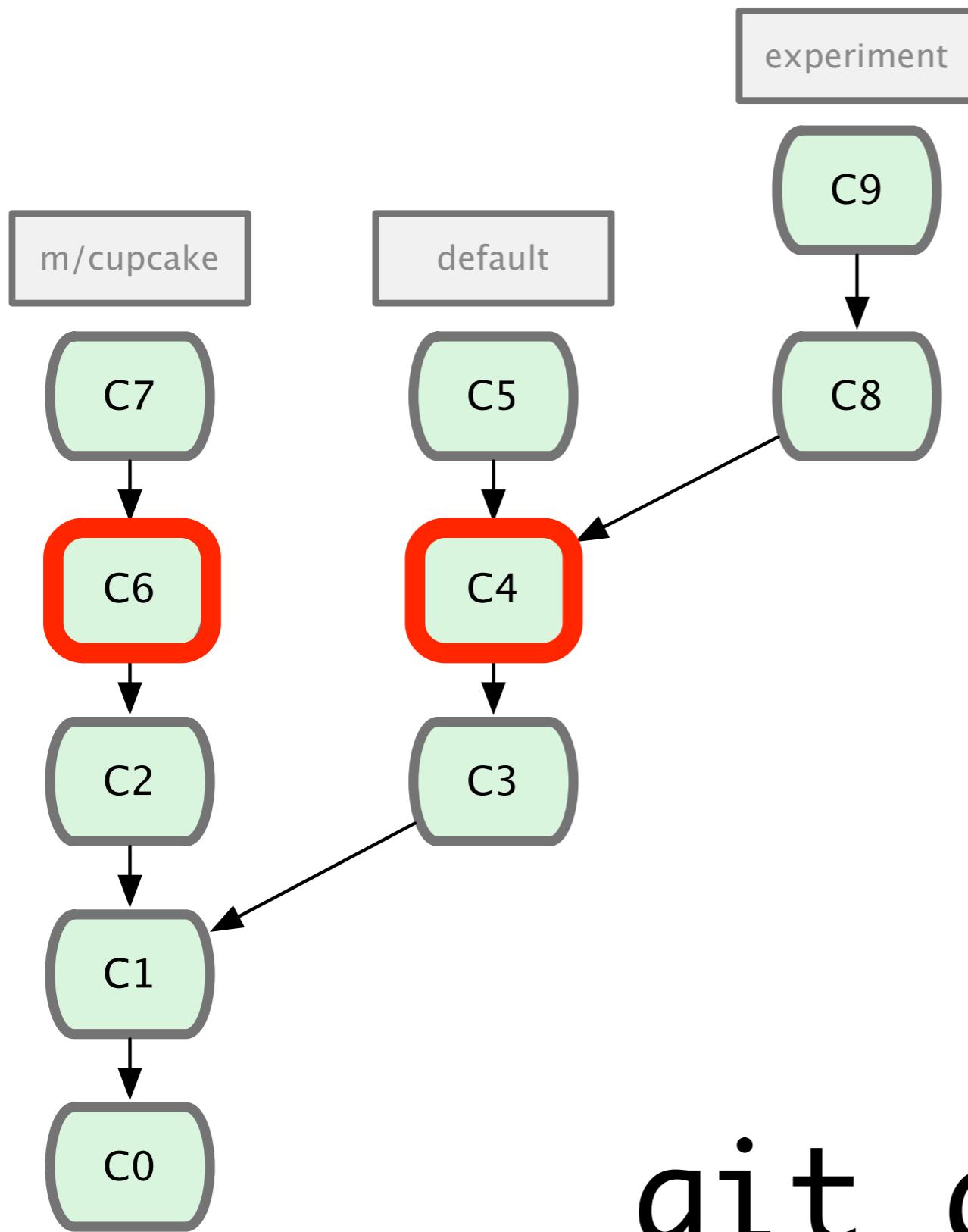




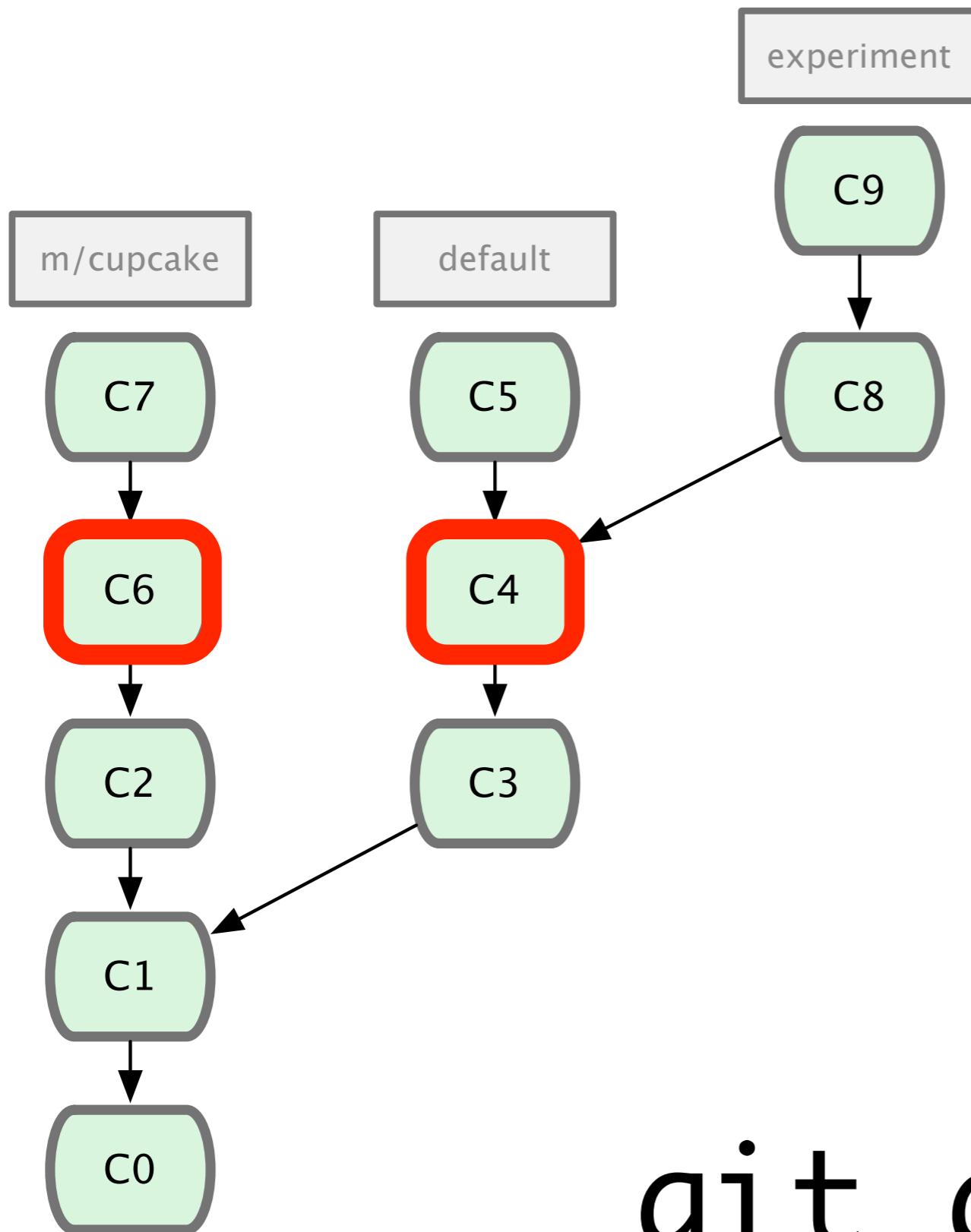
git diff c6 c4



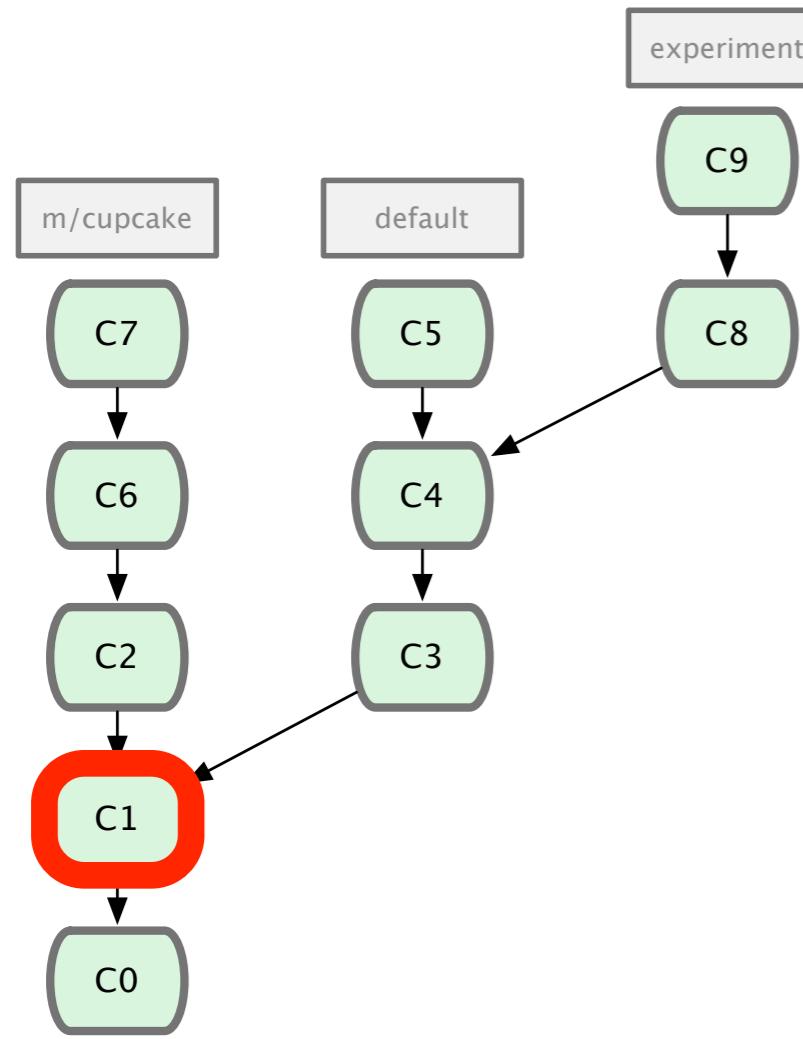
git diff c6 c4



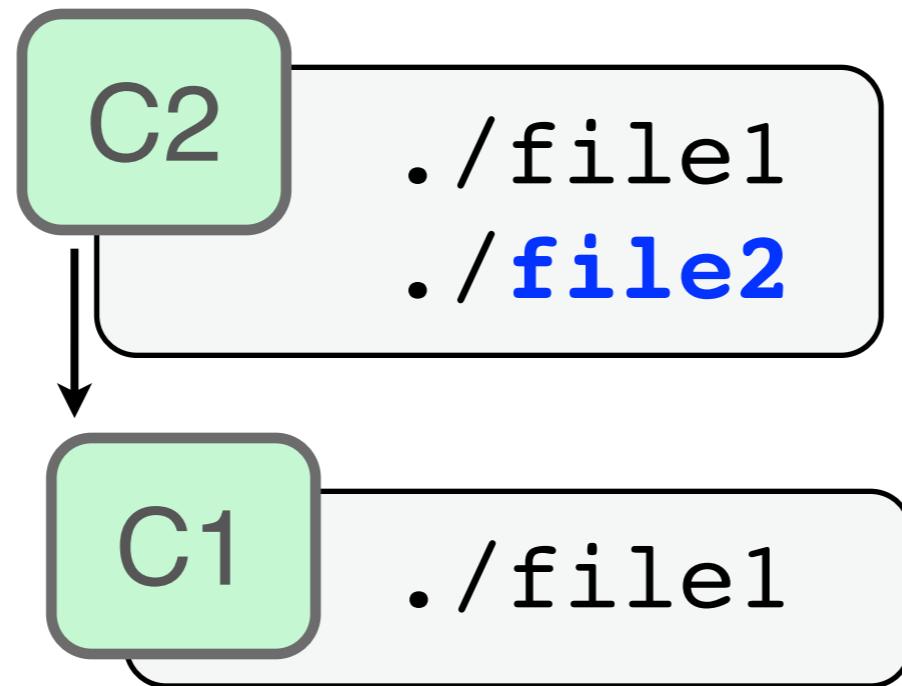
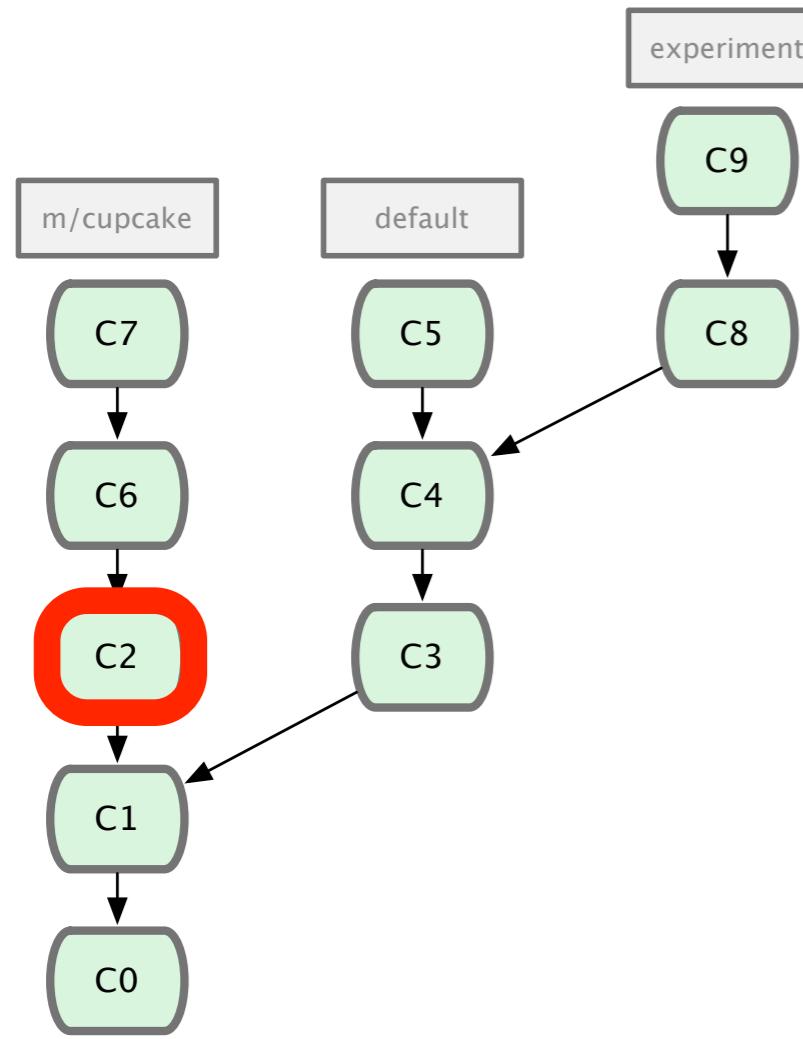
git diff c6 c4

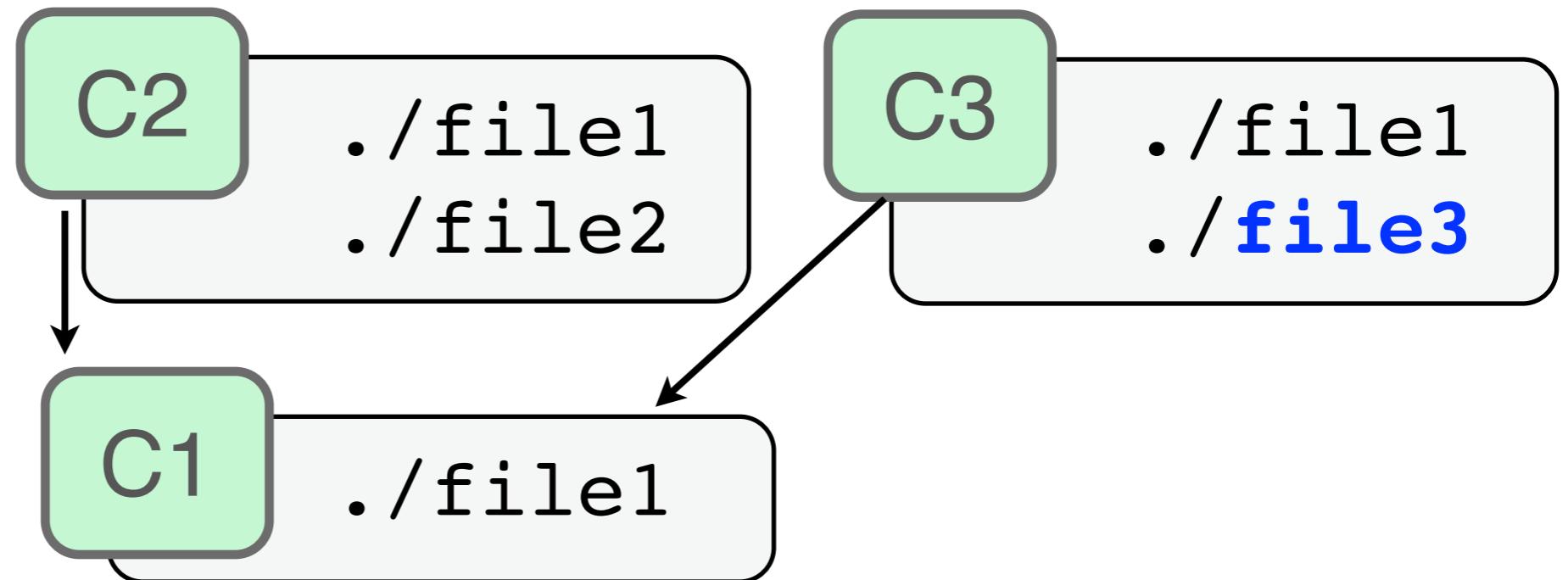
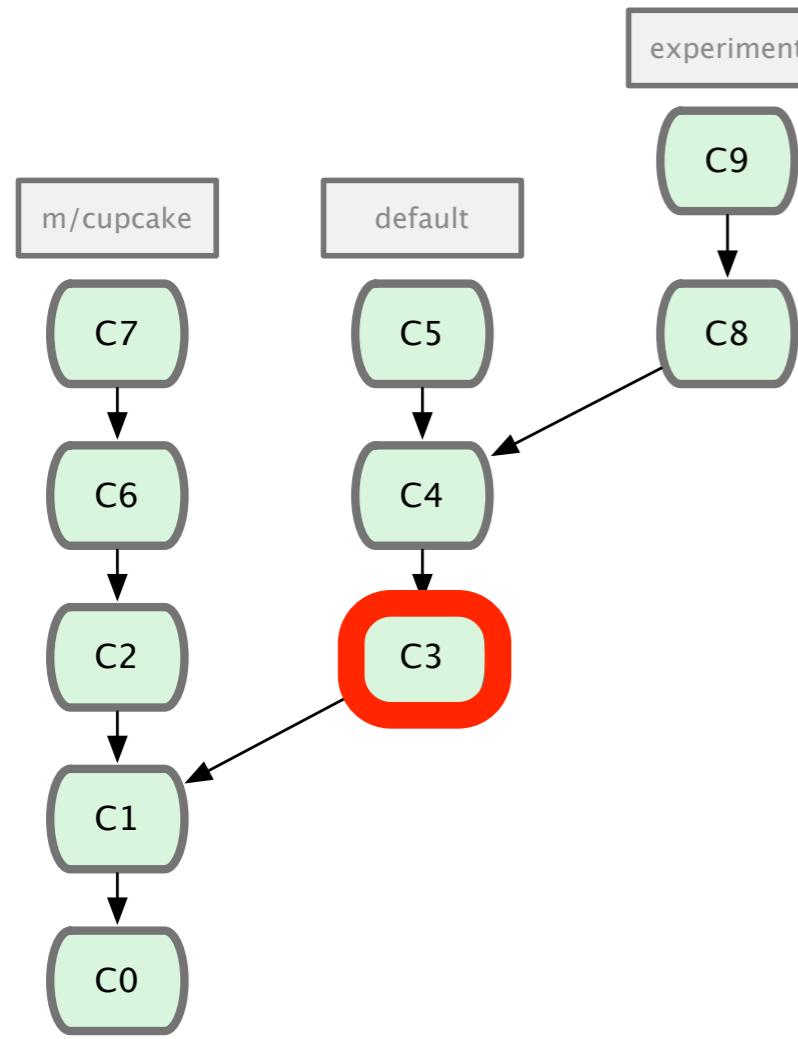


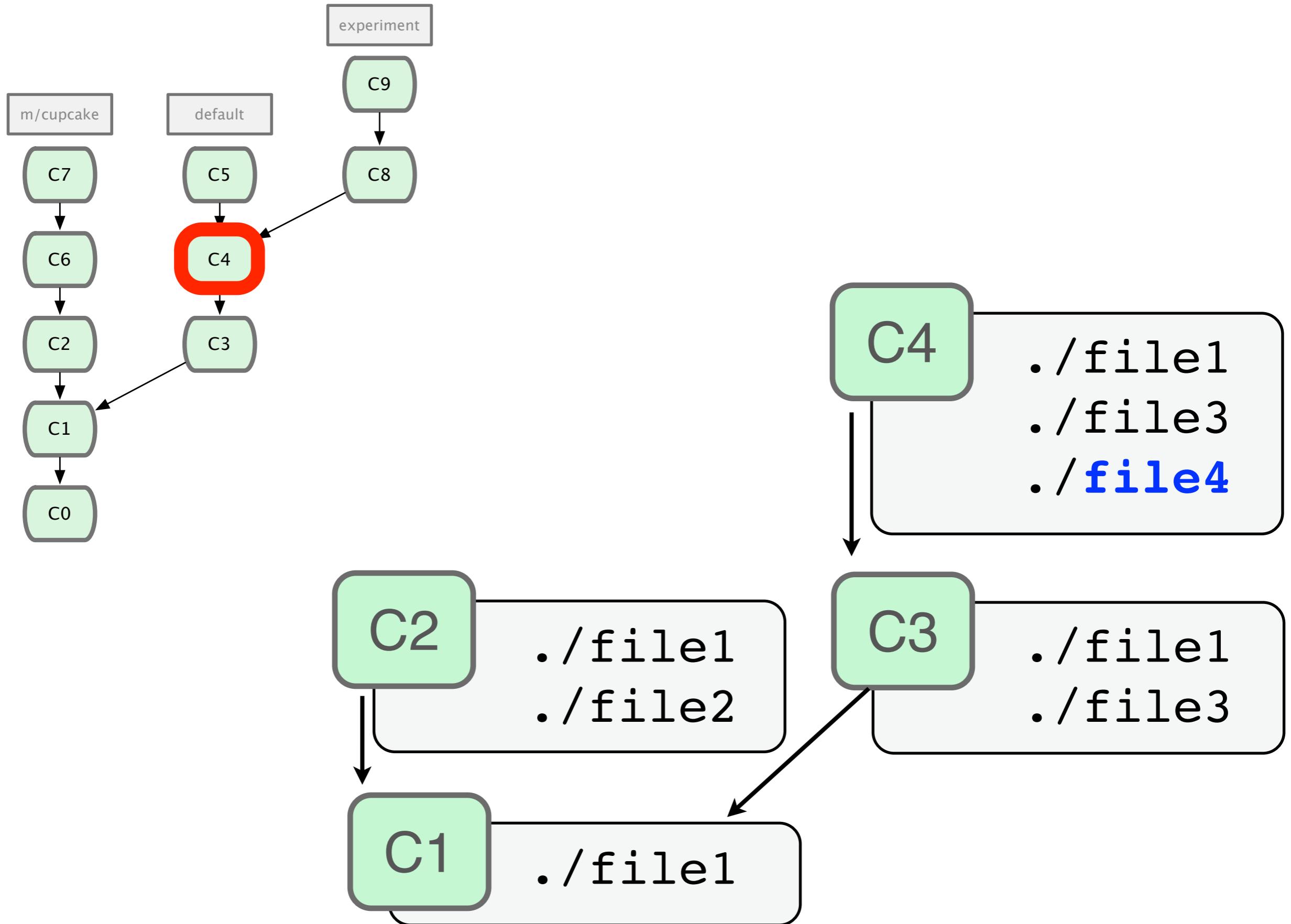
git diff c6 c4
this may not be what you want!

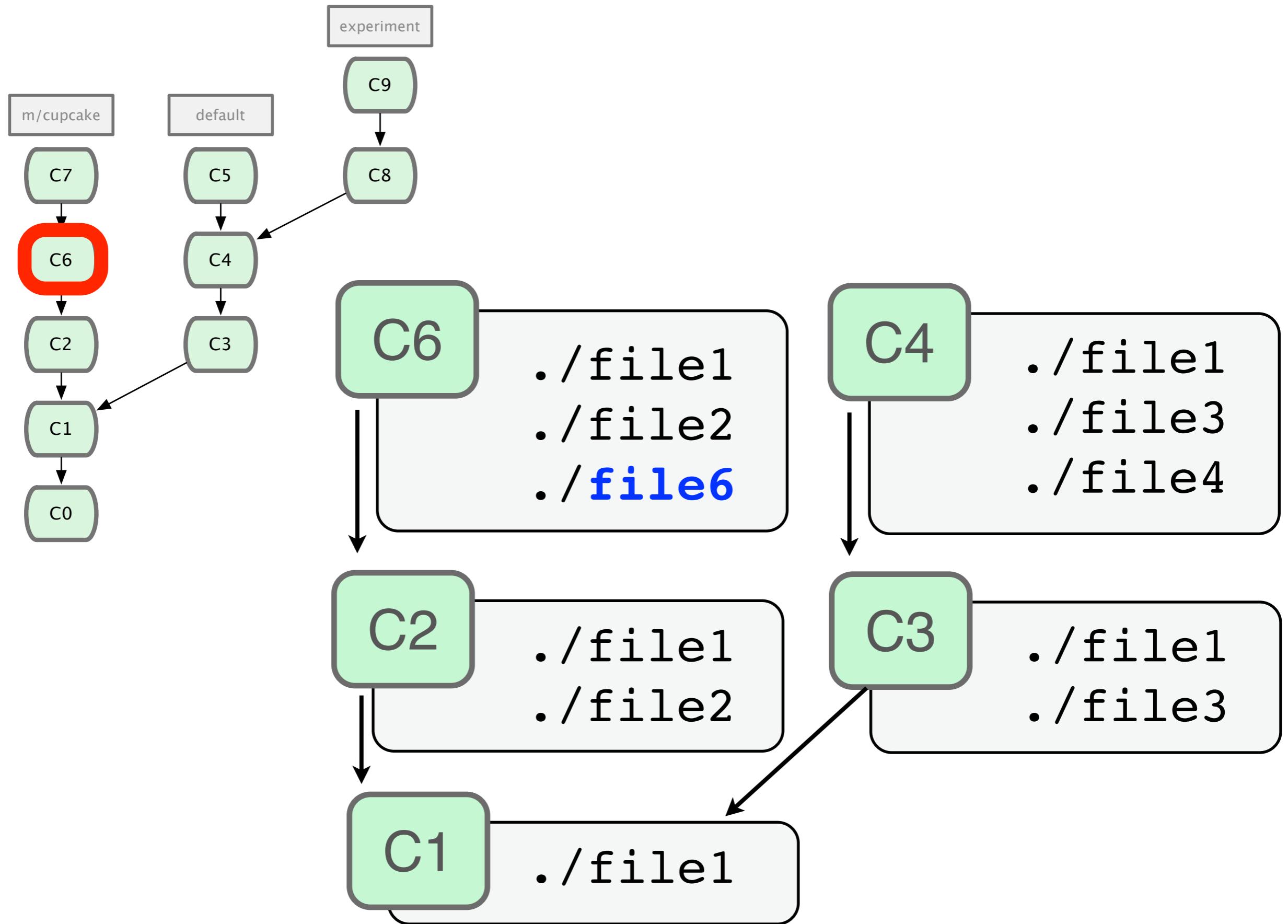


C1 ./file1



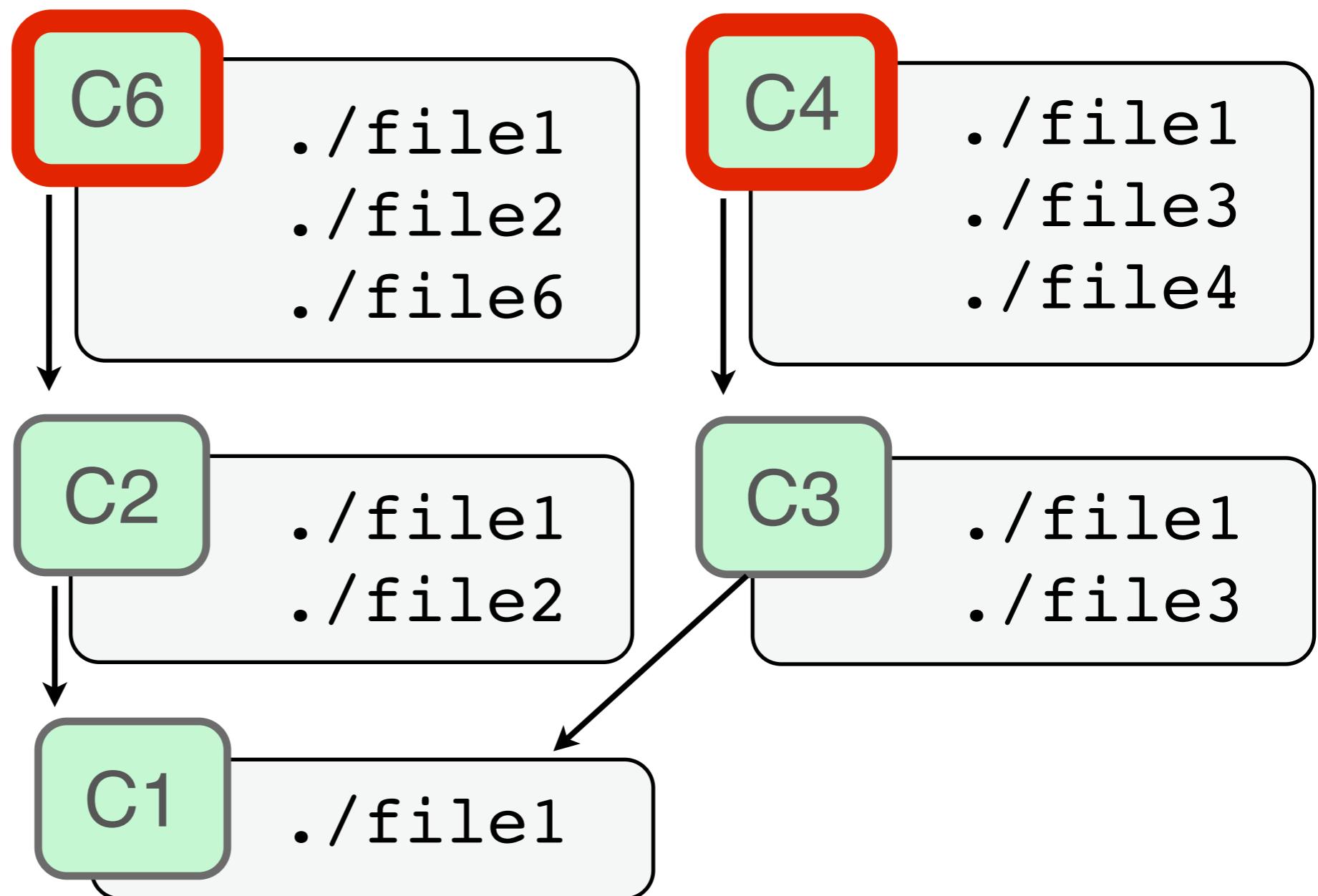


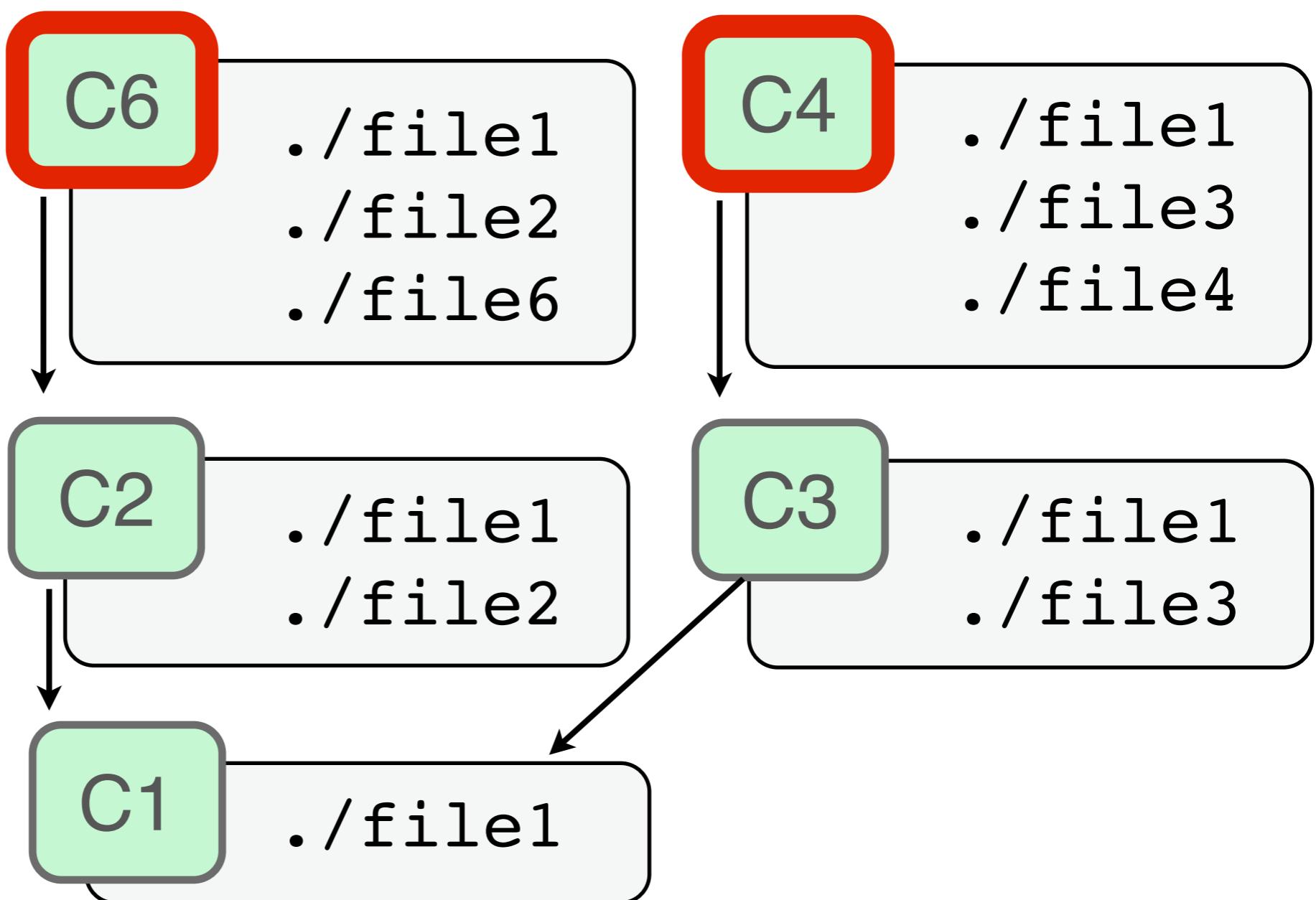
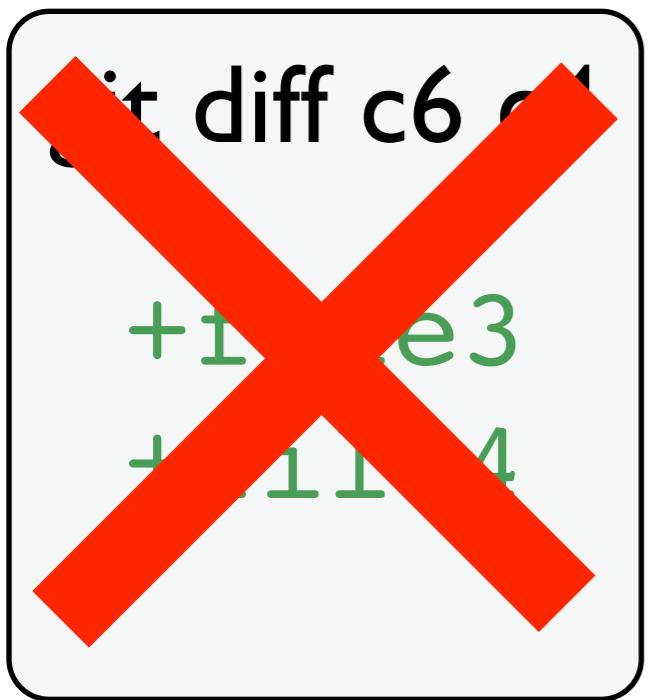




```
git diff c6 c4
```

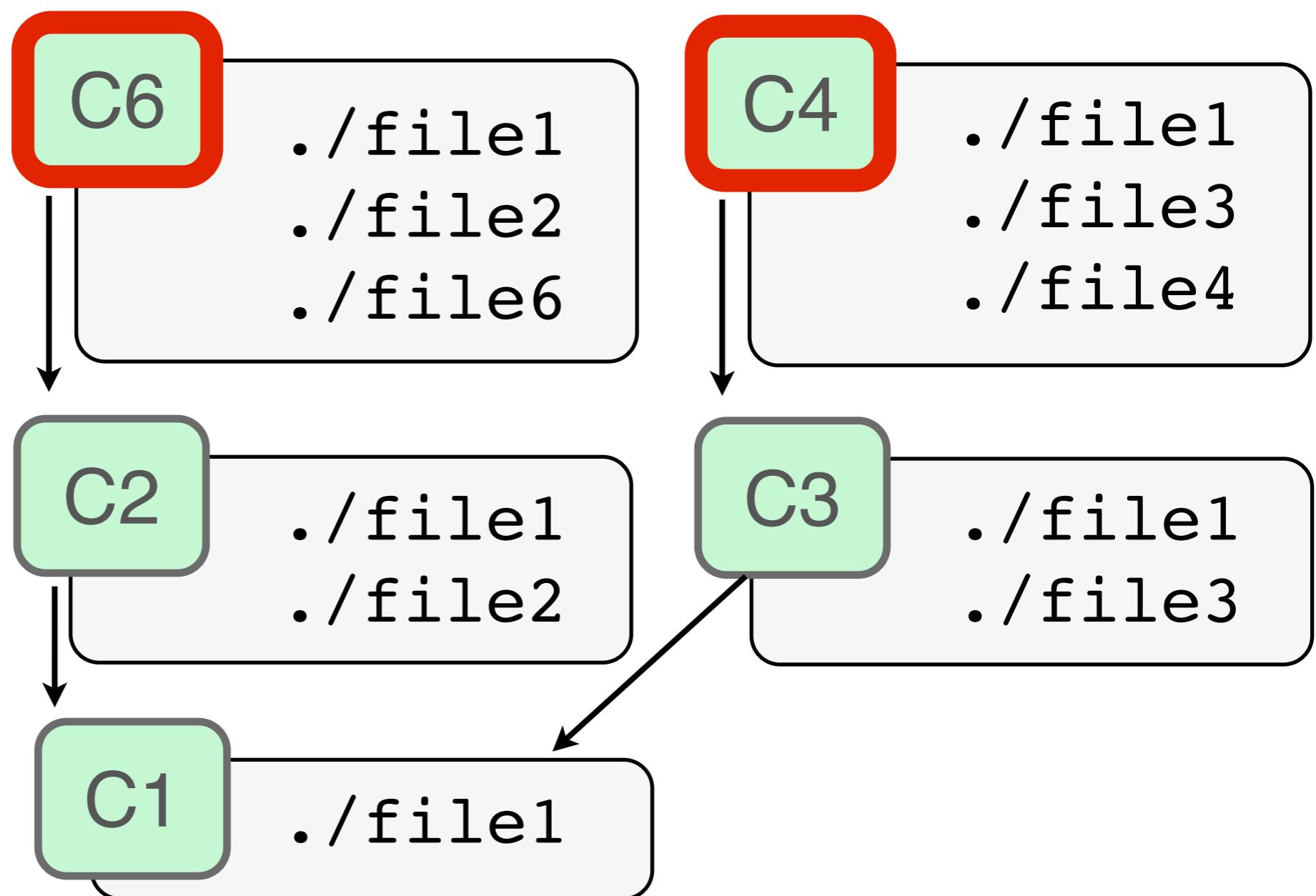
```
+file3  
+file4
```





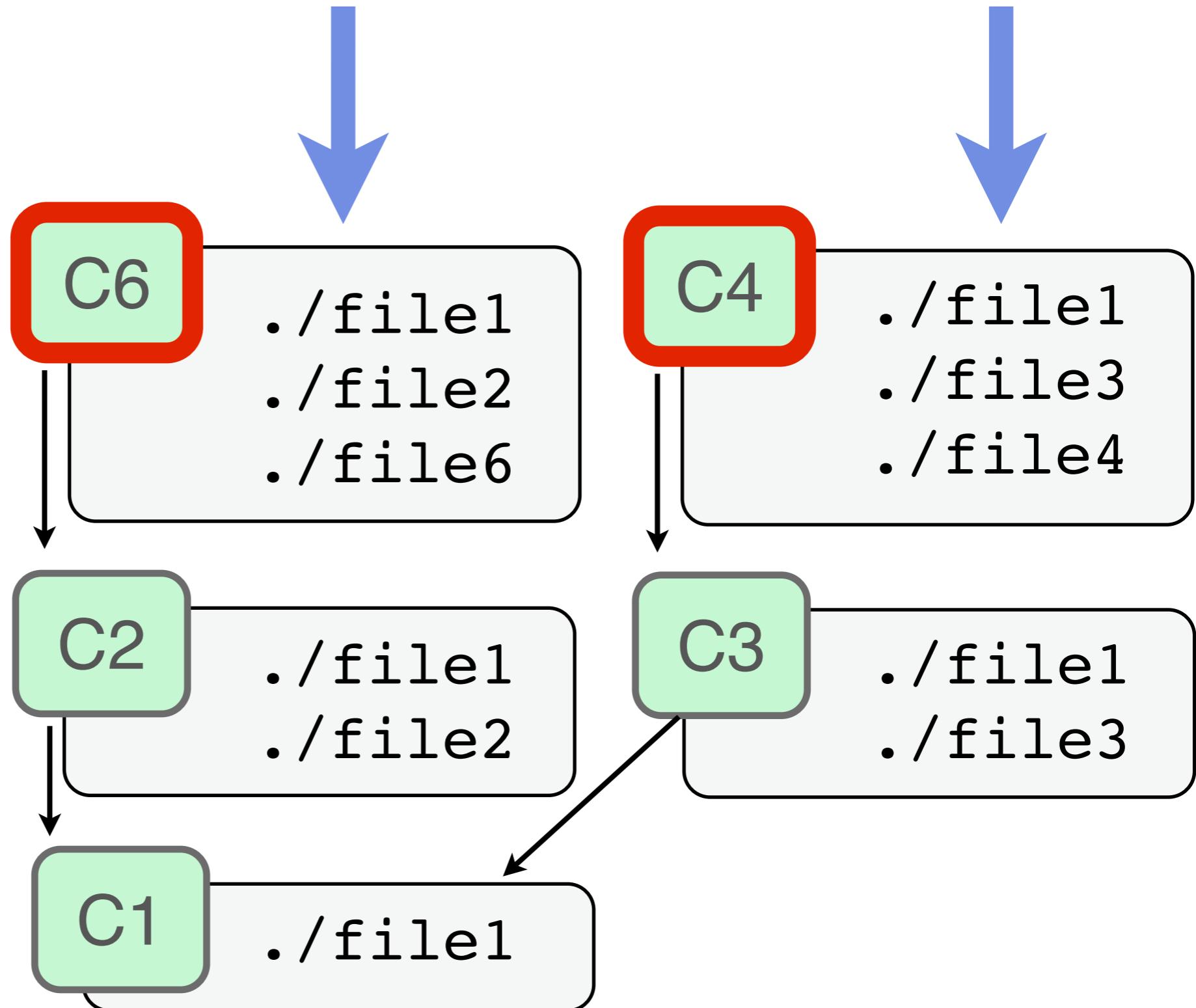
```
git diff c6 c4
```

```
+file3  
+file4  
-file2  
-file6
```



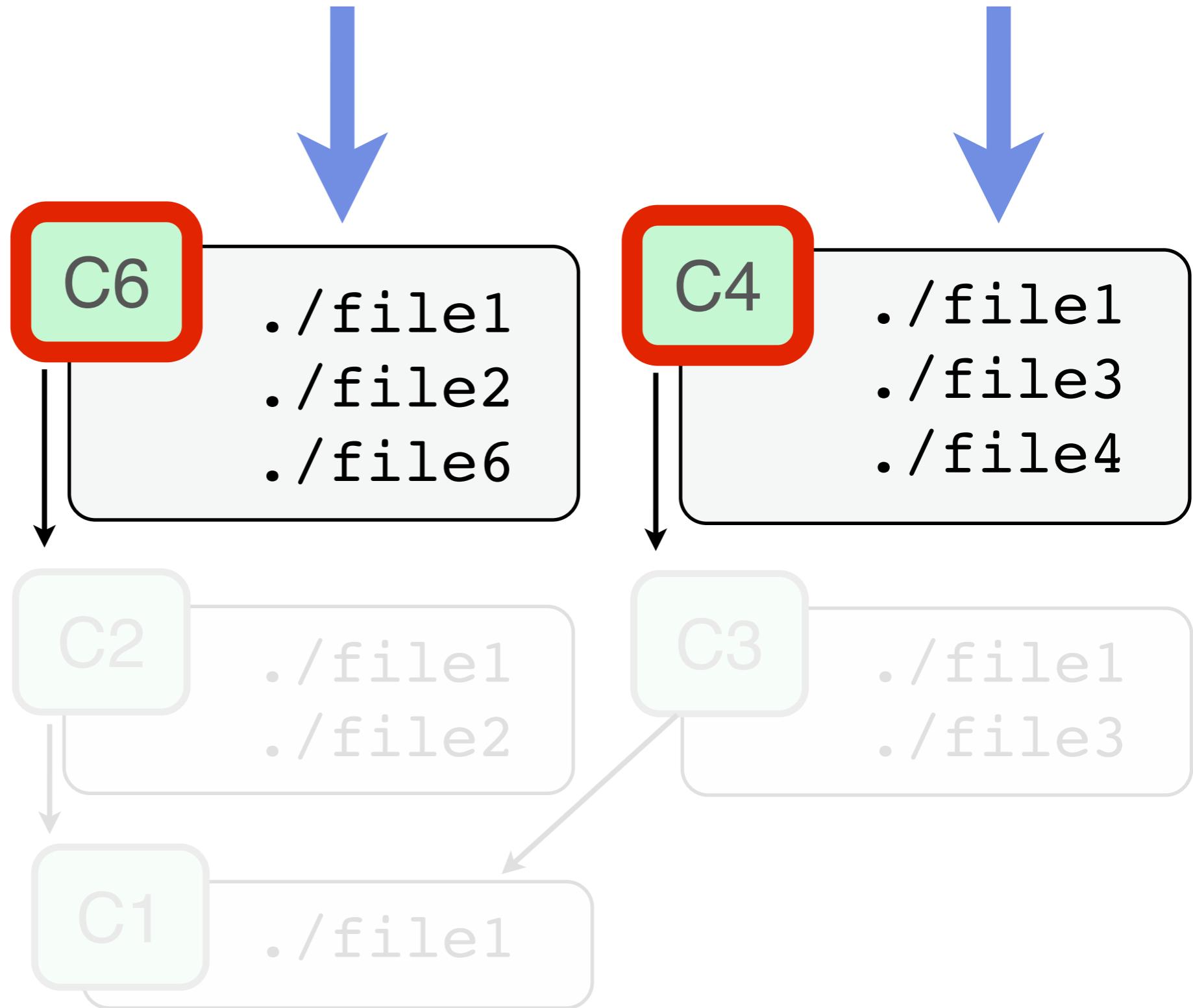
```
git diff c6 c4
```

```
+file3  
+file4  
-file2  
-file6
```



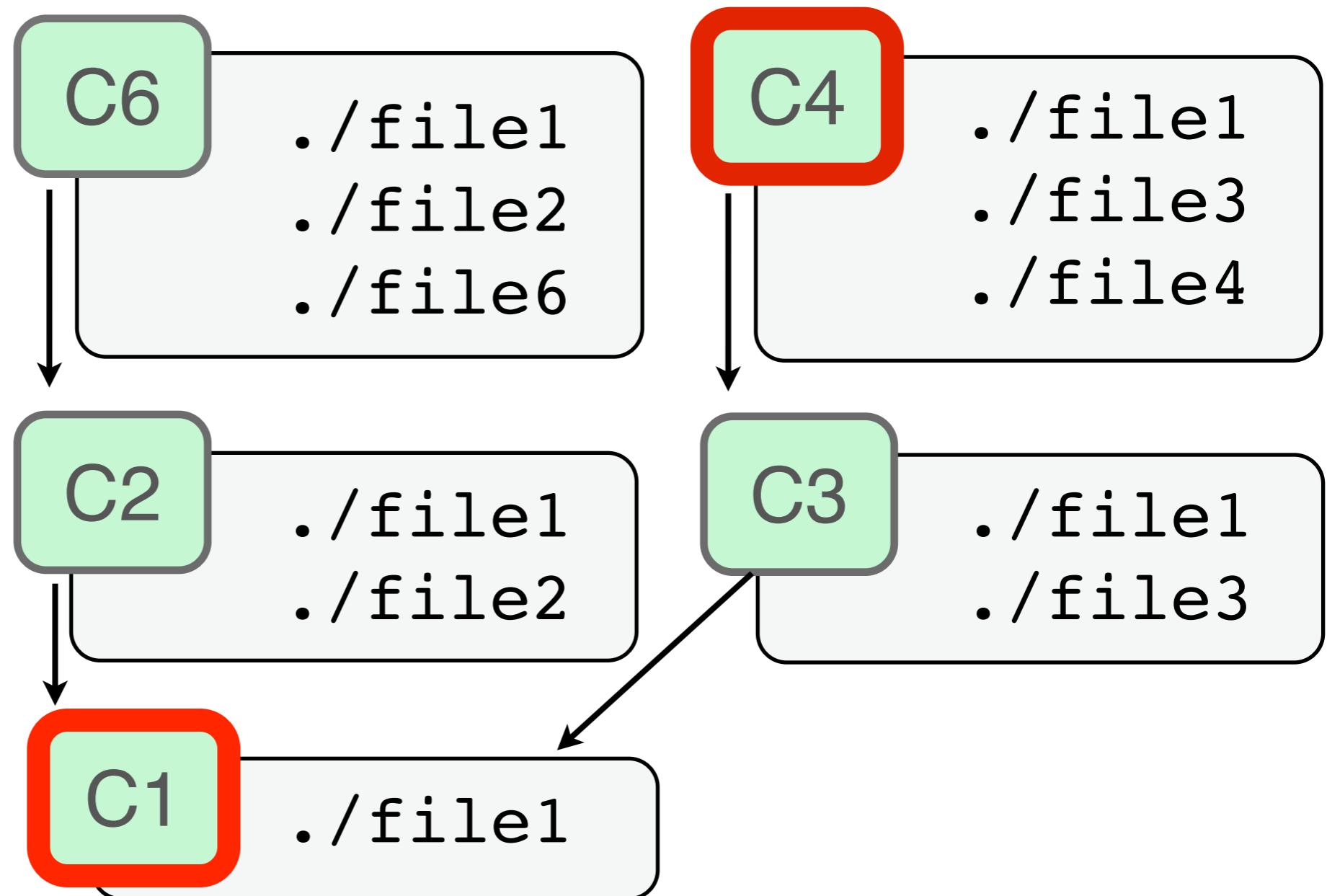
```
git diff c6 c4
```

```
+file3  
+file4  
-file2  
-file6
```



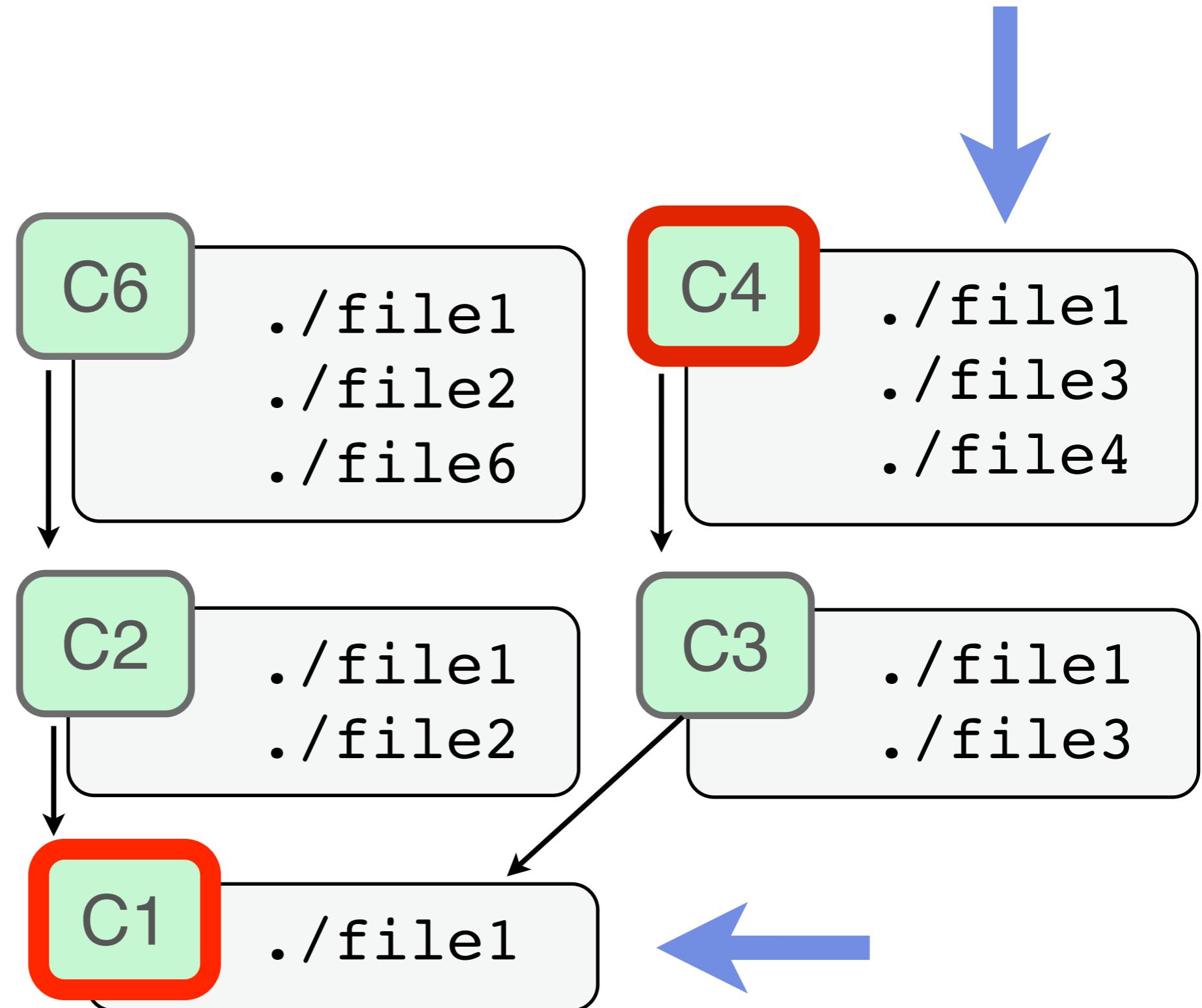
```
git diff c1 c4
```

```
+file3  
+file4
```



git diff c6...c4

+file3
+file4



Undo

Single File

git checkout [commit] file

```
$ git checkout master@{yesterday} netcfg.c
```

```
$ git checkout v1.0 netcfg.c
```

Entire Project

git reset

```
$ git status
```

```
$ git status
# Not currently on any branch.
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:   src/com/android/launcher/Workspace.java
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in
working directory)
#
# modified:   src/com/android/launcher/LiveFolder.java
#
```

```
$ git status
# Not currently on any branch.
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:   src/com/android/launcher/Workspace.java
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in
working directory)
#
# modified:   src/com/android/launcher/LiveFolder.java
#
```

```
$ git status
# Not currently on any branch.
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:   src/com/android/launcher/Workspace.java
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in
working directory)
#
# modified:   src/com/android/launcher/LiveFolder.java
#
```

by default, reset changes
the staging area

```
$ git status
# Not currently on any branch.
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:   src/com/android/launcher/Workspace.java
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in
working directory)
#
# modified:   src/com/android/launcher/LiveFolder.java
#
```

```
$ git reset --hard
```

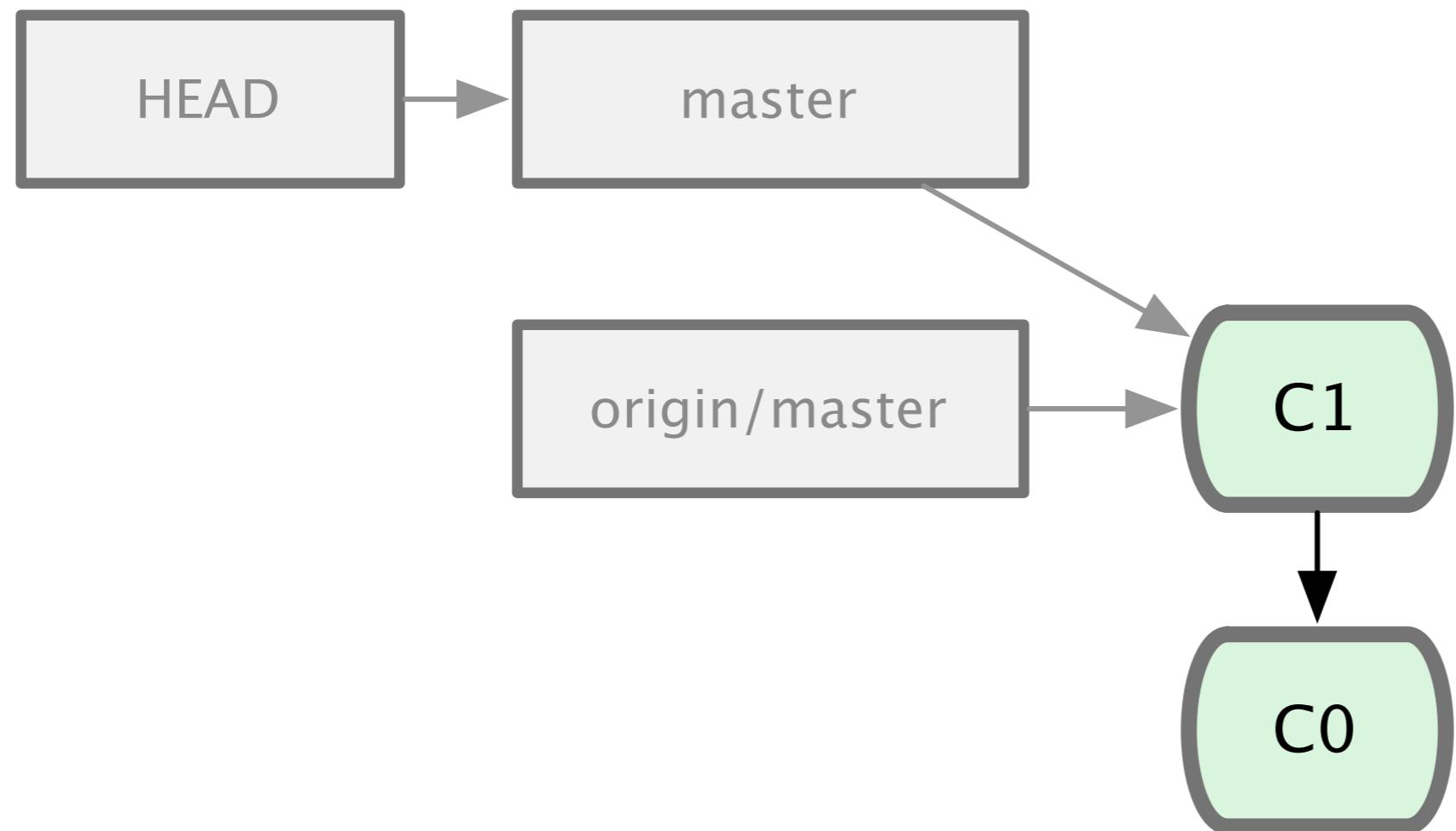
--hard changes stage
and working directory

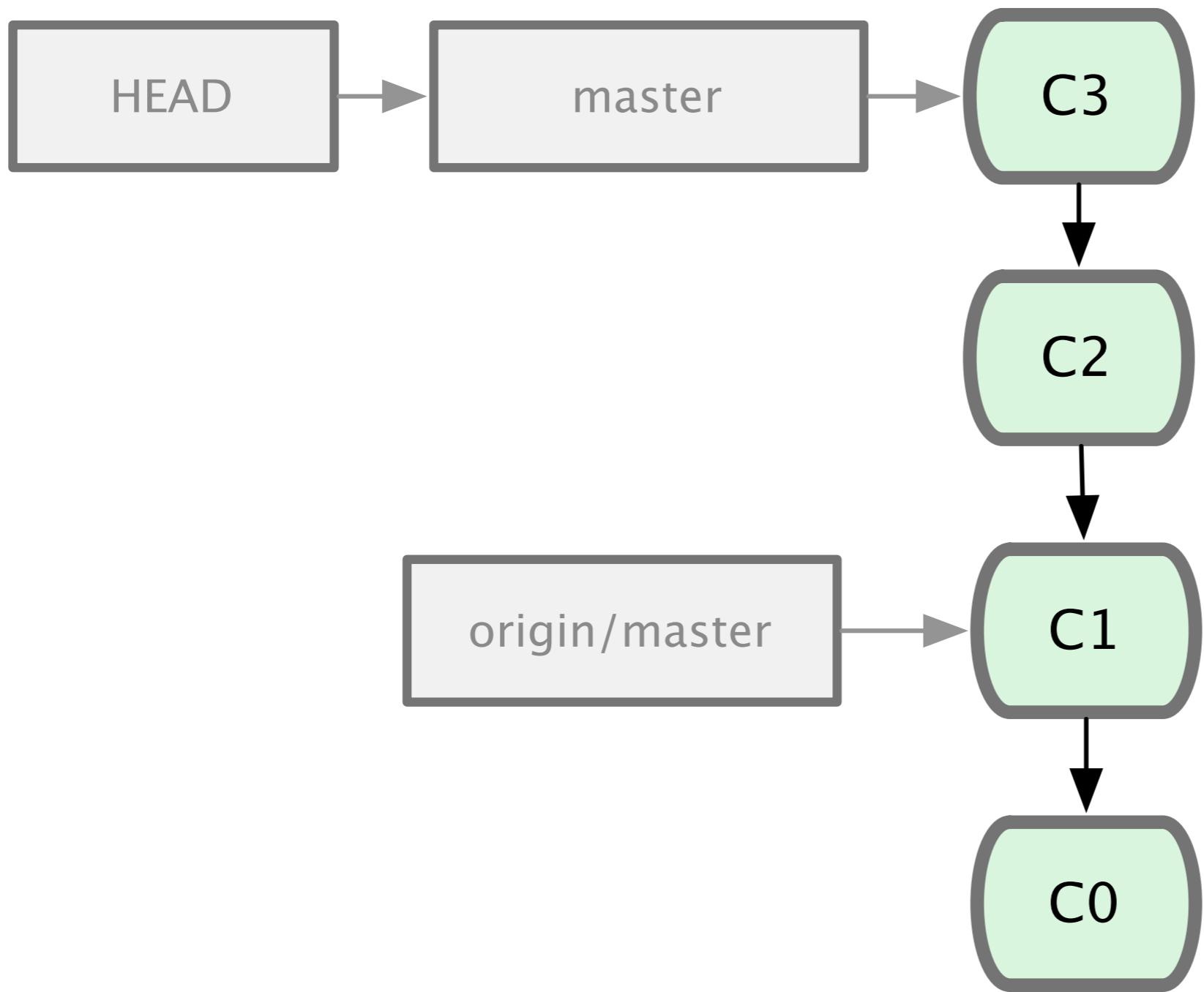
you've made commits,
but want to move them
to a topic branch

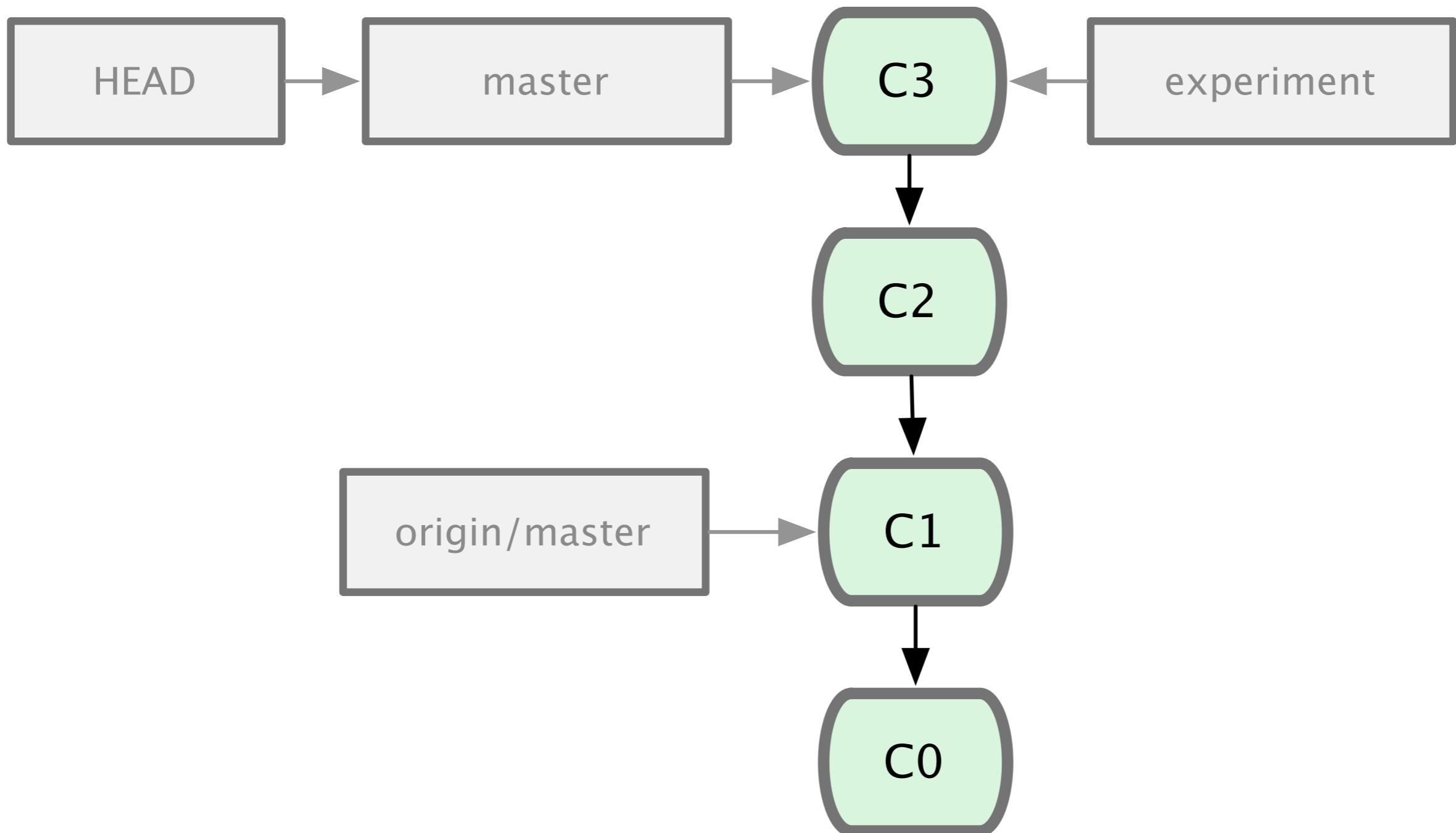
```
$ git branch experiment
```

```
$ git branch experiment  
$ git reset --hard origin/master
```

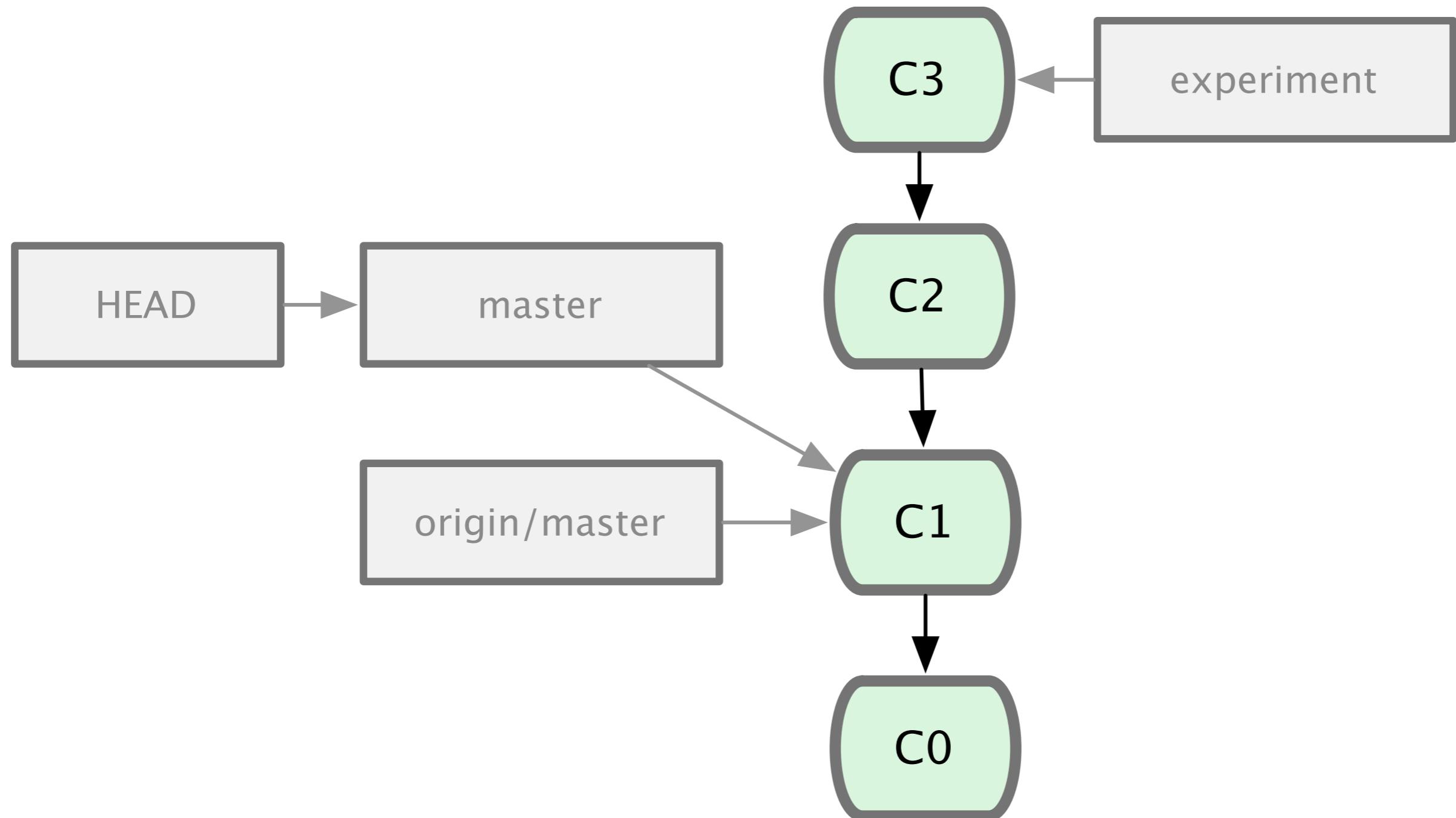
```
$ git branch experiment  
$ git reset --hard origin/master  
$ git checkout experiment
```



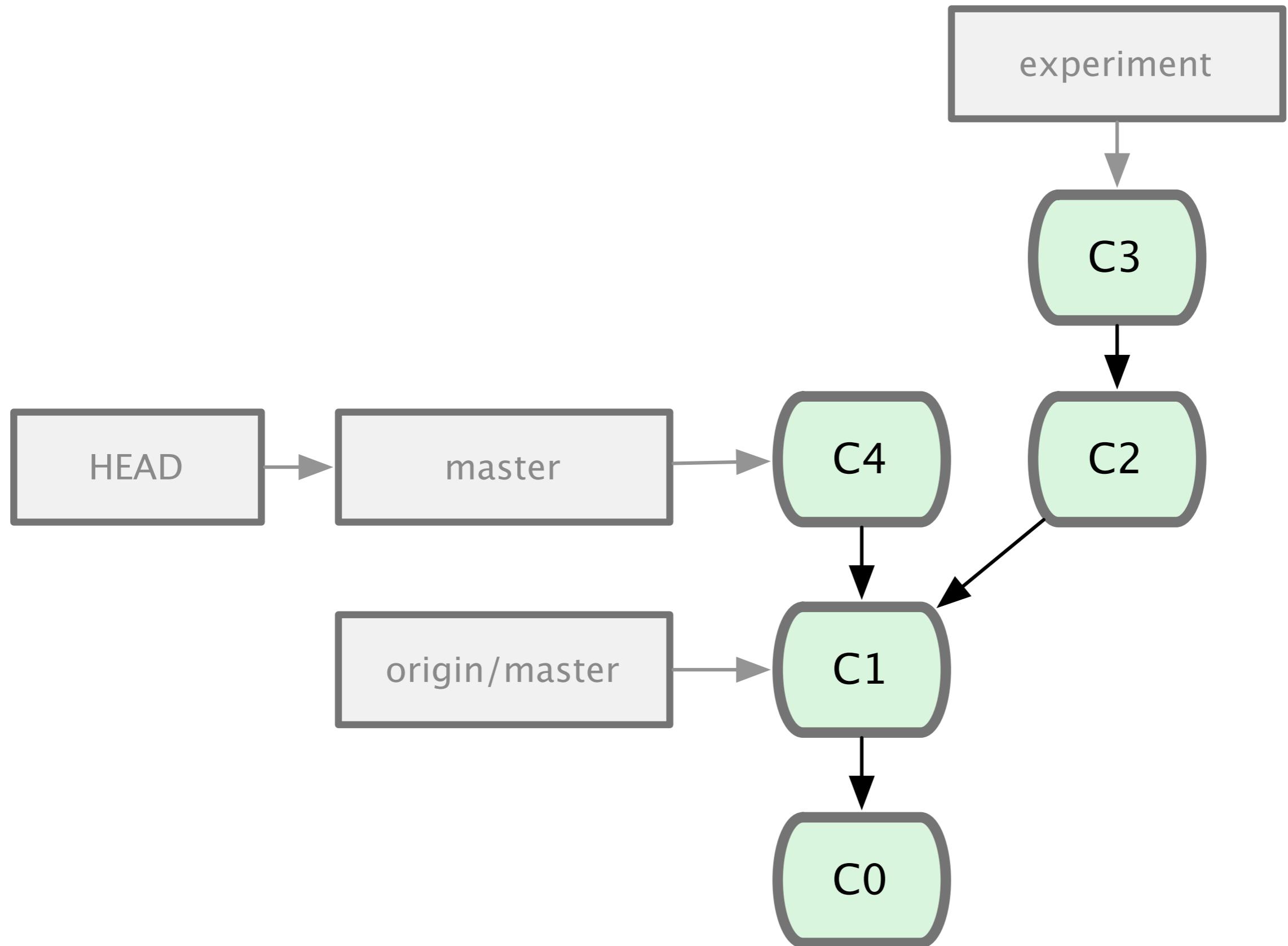




git checkout experiment



git reset --hard origin/master



git commit

Debugging

Annotation

git blame

git blame

ie: “what dumbass did this? oh, it was me...”

git blame daemon.c

979e32fa (Randal L. Schwartz	2005-10-25 16:29:09 -0700	1) #include "cache.h"
85023577 (Junio C Hamano	2006-12-19 14:34:12 -0800	2) #include "pkt-line.h"
77cb17e9 (Michal Ostrowski	2006-01-10 21:12:17 -0500	3) #include "exec_cmd.h"
49ba83fb (Jon Loeliger	2006-09-19 20:31:51 -0500	4) #include " interpolate.h"
f8ff0c06 (Petr Baudis	2005-09-22 11:25:28 +0200	5)
85023577 (Junio C Hamano	2006-12-19 14:34:12 -0800	6) #include <syslog.h>
85023577 (Junio C Hamano	2006-12-19 14:34:12 -0800	7)
695dffe2 (Johannes Schindelin	2006-09-28 12:00:35 +0200	8) #ifndef HOST_NAME_MAX
695dffe2 (Johannes Schindelin	2006-09-28 12:00:35 +0200	9) #define HOST_NAME_MAX 256
695dffe2 (Johannes Schindelin	2006-09-28 12:00:35 +0200	10) #endif
695dffe2 (Johannes Schindelin	2006-09-28 12:00:35 +0200	11)
415e7b87 (Patrick Welche	2007-10-18 18:17:39 +0100	12) #ifndef NI_MAXSERV
415e7b87 (Patrick Welche	2007-10-18 18:17:39 +0100	13) #define NI_MAXSERV 32
415e7b87 (Patrick Welche	2007-10-18 18:17:39 +0100	14) #endif
415e7b87 (Patrick Welche	2007-10-18 18:17:39 +0100	15)
9048fe1c (Petr Baudis	2005-09-24 16:13:01 +0200	16) static int log_syslog;
f8ff0c06 (Petr Baudis	2005-09-22 11:25:28 +0200	17) static int verbose;
1955fabf (Mark Wooding	2006-02-03 20:27:04 +0000	18) static int reuseaddr;
f8ff0c06 (Petr Baudis	2005-09-22 11:25:28 +0200	19)
960deccb (H. Peter Anvin	2005-10-19 14:27:01 -0700	20) static const char daemon_usage[] =
1b1dd23f (Stephan Beyer	2008-07-13 15:36:15 +0200	21) "git daemon [--verbose] [--syslog] [
3bd62c21 (Stephen R. van den Berg	2008-08-14 20:02:20 +0200	22) " --timeout=n] [--init-ti
3bd62c21 (Stephen R. van den Berg	2008-08-14 20:02:20 +0200	23) " --strict-paths] [--base
73a7a656 (Jens Axboe	2007-07-27 14:00:29 -0700	24) " --user-path --user-pa
49ba83fb (Jon Loeliger	2006-09-19 20:31:51 -0500	25) " --interpolated-path=pat
678dac6b (Tilman Sauerbeck	2006-08-22 19:37:41 +0200	26) " --reuseaddr] [--detach]
d9edcbd6 (Junio C Hamano	2006-09-07 01:40:04 -0700	27) " --[enable disable allow
dd467629 (Jon Loeliger	2006-09-26 09:47:43 -0500	28) " --inetd [--listen=hos
dd467629 (Jon Loeliger	2006-09-26 09:47:43 -0500	29) " --user=user
dd467629 (Jon Loeliger	2006-09-26 09:47:43 -0500	30) " [directory...]";
4ae95682 (H. Peter Anvin	2005-09-26 19:10:55 -0700	31)
4ae95682 (H. Peter Anvin	2005-09-26 19:10:55 -0700	32) /* List of acceptable pathname prefi
96f1e58f (David Rienties	2006-08-15 10:23:48 -0700	33) static char **ok_paths:

git blame daemon.c

979e32fa	(Randal L. Schwartz	2005-10-25 16:29:09 -0700	1) #include "cache.h"
85023577	(Junio C Hamano	2006-12-19 14:34:12 -0800	2) #include "pkt-line.h"
77cb17e9	(Michal Ostrowski	2006-01-10 21:12:17 -0500	3) #include "exec_cmd.h"
49ba83fb	(Jon Loeliger	2006-09-19 20:31:51 -0500	4) #include " interpolate.h"
f8ff0c06	(Petr Baudis	2005-09-22 11:25:28 +0200	5)
85023577	(Junio C Hamano	2006-12-19 14:34:12 -0800	6) #include <syslog.h>
85023577	(Junio C Hamano	2006-12-19 14:34:12 -0800	7)
695dffe2	(Johannes Schindelin	2006-09-28 12:00:35 +0200	8) #ifndef HOST_NAME_MAX
695dffe2	(Johannes Schindelin	2006-09-28 12:00:35 +0200	9) #define HOST_NAME_MAX 256
695dffe2	(Johannes Schindelin	2006-09-28 12:00:35 +0200	10) #endif
695dffe2	(Johannes Schindelin	2006-09-28 12:00:35 +0200	11)
415e7b87	(Patrick Welche	2007-10-18 18:17:39 +0100	12) #ifndef NI_MAXSERV
415e7b87	(Patrick Welche	2007-10-18 18:17:39 +0100	13) #define NI_MAXSERV 32
415e7b87	(Patrick Welche	2007-10-18 18:17:39 +0100	14) #endif
415e7b87	(Patrick Welche	2007-10-18 18:17:39 +0100	15)
9048fe1c	(Petr Baudis	2005-09-24 16:13:01 +0200	16) static int log_syslog;
f8ff0c06	(Petr Baudis	2005-09-22 11:25:28 +0200	17) static int verbose;
1955fabf	(Mark Wooding	2006-02-03 20:27:04 +0000	18) static int reuseaddr;
f8ff0c06	(Petr Baudis	2005-09-22 11:25:28 +0200	19)
960deccb	(H. Peter Anvin	2005-10-19 14:27:01 -0700	20) static const char daemon_usage[] =
1b1dd23f	(Stephan Beyer	2008-07-13 15:36:15 +0200	21) "git daemon [--verbose] [--syslog] [
3bd62c21	(Stephen R. van den Berg	2008-08-14 20:02:20 +0200	22) " --timeout=n] [--init-ti
3bd62c21	(Stephen R. van den Berg	2008-08-14 20:02:20 +0200	23) " --strict-paths] [--base
73a7a656	(Jens Axboe	2007-07-27 14:00:29 -0700	24) " --user-path --user-pa
49ba83fb	(Jon Loeliger	2006-09-19 20:31:51 -0500	25) " --interpolated-path=pat
678dac6b	(Tilman Sauerbeck	2006-08-22 19:37:41 +0200	26) " --reuseaddr] [--detach]
d9edcbd6	(Junio C Hamano	2006-09-07 01:40:04 -0700	27) " --[enable disable allow
dd467629	(Jon Loeliger	2006-09-26 09:47:43 -0500	28) " --inetd [--listen=hos
dd467629	(Jon Loeliger	2006-09-26 09:47:43 -0500	29) " --user=user
dd467629	(Jon Loeliger	2006-09-26 09:47:43 -0500	30) " [directory...]";
4ae95682	(H. Peter Anvin	2005-09-26 19:10:55 -0700	31)
4ae95682	(H. Peter Anvin	2005-09-26 19:10:55 -0700	32) /* List of acceptable pathname prefi
96f1e58f	(David Rienties	2006-08-15 10:23:48 -0700	33) static char **ok_paths:

git blame daemon.c

979e32fa	(Randal L. Schwartz	2005-10-25 16:29:09 -0700	1) #include "cache.h"
85023577	(Junio C Hamano	2006-12-19 14:34:12 -0800	2) #include "pkt-line.h"
77cb17e9	(Michal Ostrowski	2006-01-10 21:12:17 -0500	3) #include "exec_cmd.h"
49ba83fb	(Jon Loeliger	2006-09-19 20:31:51 -0500	4) #include " interpolate.h"
f8ff0c06	(Petr Baudis	2005-09-22 11:25:28 +0200	5)
85023577	(Junio C Hamano	2006-12-19 14:34:12 -0800	6) #include <syslog.h>
85023577	(Junio C Hamano	2006-12-19 14:34:12 -0800	7)
695dffe2	(Johannes Schindelin	2006-09-28 12:00:35 +0200	8) #ifndef HOST_NAME_MAX
695dffe2	(Johannes Schindelin	2006-09-28 12:00:35 +0200	9) #define HOST_NAME_MAX 256
695dffe2	(Johannes Schindelin	2006-09-28 12:00:35 +0200	10) #endif
695dffe2	(Johannes Schindelin	2006-09-28 12:00:35 +0200	11)
415e7b87	(Patrick Welche	2007-10-18 18:17:39 +0100	12) #ifndef NI_MAXSERV
415e7b87	(Patrick Welche	2007-10-18 18:17:39 +0100	13) #define NI_MAXSERV 32
415e7b87	(Patrick Welche	2007-10-18 18:17:39 +0100	14) #endif
415e7b87	(Patrick Welche	2007-10-18 18:17:39 +0100	15)
9048fe1c	(Petr Baudis	2005-09-24 16:13:01 +0200	16) static int log_syslog;
f8ff0c06	(Petr Baudis	2005-09-22 11:25:28 +0200	17) static int verbose;
1955fabf	(Mark Wooding	2006-02-03 20:27:04 +0000	18) static int reuseaddr;
f8ff0c06	(Petr Baudis	2005-09-22 11:25:28 +0200	19)
960deccb	(H. Peter Anvin	2005-10-19 14:27:01 -0700	20) static const char daemon_usage[] =
1b1dd23f	(Stephan Beyer	2008-07-13 15:36:15 +0200	21) "git daemon [--verbose] [--syslog] [
3bd62c21	(Stephen R. van den Berg	2008-08-14 20:02:20 +0200	22) " --timeout=n] [--init-ti
3bd62c21	(Stephen R. van den Berg	2008-08-14 20:02:20 +0200	23) " --strict-paths] [--base
73a7a656	(Jens Axboe	2007-07-27 14:00:29 -0700	24) " --user-path --user-pa
49ba83fb	(Jon Loeliger	2006-09-19 20:31:51 -0500	25) " --interpolated-path=pat
678dac6b	(Tilman Sauerbeck	2006-08-22 19:37:41 +0200	26) " --reuseaddr] [--detach]
d9edcbd6	(Junio C Hamano	2006-09-07 01:40:04 -0700	27) " --[enable disable allow
dd467629	(Jon Loeliger	2006-09-26 09:47:43 -0500	28) " --inetd [--listen=hos
dd467629	(Jon Loeliger	2006-09-26 09:47:43 -0500	29) " [--user=user
dd467629	(Jon Loeliger	2006-09-26 09:47:43 -0500	30) " [directory...]";
4ae95682	(H. Peter Anvin	2005-09-26 19:10:55 -0700	31)
4ae95682	(H. Peter Anvin	2005-09-26 19:10:55 -0700	32) /* List of acceptable pathname prefi
96f1e58f	(David Rientjes	2006-08-15 10:23:48 -0700	33) static char **ok_paths:

git blame daemon.c

```
979e32fa (Randal L. Schwartz
85023577 (Junio C Hamano
77cb17e9 (Michal Ostrowski
49ba83fb (Jon Loeliger
f8ff0c06 (Petr Baudis
85023577 (Junio C Hamano
85023577 (Junio C Hamano
695dffe2 (Johannes Schindelin
695dffe2 (Johannes Schindelin
695dffe2 (Johannes Schindelin
695dffe2 (Johannes Schindelin
415e7b87 (Patrick Welche
415e7b87 (Patrick Welche
415e7b87 (Patrick Welche
415e7b87 (Patrick Welche
9048fe1c (Petr Baudis
f8ff0c06 (Petr Baudis
1955fabf (Mark Wooding
f8ff0c06 (Petr Baudis
960deccb (H. Peter Anvin
1b1dd23f (Stephan Beyer
3bd62c21 (Stephen R. van den Berg
3bd62c21 (Stephen R. van den Berg
73a7a656 (Jens Axboe
49ba83fb (Jon Loeliger
678dac6b (Tilman Sauerbeck
d9edcbd6 (Junio C Hamano
dd467629 (Jon Loeliger
dd467629 (Jon Loeliger
dd467629 (Jon Loeliger
4ae95682 (H. Peter Anvin
4ae95682 (H. Peter Anvin
96f1e58f (David Rientjes
```

```
2005-10-25 16:29:09 -0700
2006-12-19 14:34:12 -0800
2006-01-10 21:12:17 -0500
2006-09-19 20:31:51 -0500
2005-09-22 11:25:28 +0200
2006-12-19 14:34:12 -0800
2006-12-19 14:34:12 -0800
2006-09-28 12:00:35 +0200
2006-09-28 12:00:35 +0200
2006-09-28 12:00:35 +0200
2006-09-28 12:00:35 +0200
2007-10-18 18:17:39 +0100
2007-10-18 18:17:39 +0100
2007-10-18 18:17:39 +0100
2007-10-18 18:17:39 +0100
2005-09-24 16:13:01 +0200
2005-09-22 11:25:28 +0200
2006-02-03 20:27:04 +0000
2005-09-22 11:25:28 +0200
2005-10-19 14:27:01 -0700
2008-07-13 15:36:15 +0200
2008-08-14 20:02:20 +0200
2008-08-14 20:02:20 +0200
2007-07-27 14:00:29 -0700
2006-09-19 20:31:51 -0500
2006-08-22 19:37:41 +0200
2006-09-07 01:40:04 -0700
2006-09-26 09:47:43 -0500
2006-09-26 09:47:43 -0500
2006-09-26 09:47:43 -0500
2005-09-26 19:10:55 -0700
2005-09-26 19:10:55 -0700
2006-08-15 10:23:48 -0700
```

```
1) #include "cache.h"
2) #include "pkt-line.h"
3) #include "exec_cmd.h"
4) #include " interpolate.h"
5)
6) #include <syslog.h>
7)
8) #ifndef HOST_NAME_MAX
9) #define HOST_NAME_MAX 256
10) #endif
11)
12) #ifndef NI_MAXSERV
13) #define NI_MAXSERV 32
14) #endif
15)
16) static int log_syslog;
17) static int verbose;
18) static int reuseaddr;
19)
20) static const char daemon_usage[] =
21) "git daemon [--verbose] [--syslog] [
22) " --timeout=n ] [--init-ti
23) " --strict-paths] [--base
24) " --user-path | --user-pa
25) " --interpolated-path=pat
26) " --reuseaddr] [--detach]
27) " [--enable|disable|allow
28) " --inetd | [--listen=hos
29) " --user=user
30) " [directory...]";
31)
32) /* List of acceptable pathname prefi
33) static char **ok_paths;
```

git blame daemon.c

```
979e32fa (Randal L. Schwartz 2005-10-25 16:29:09 -0700
85023577 (Junio C Hamano 2006-12-19 14:34:12 -0800
77cb17e9 (Michal Ostrowski 2006-01-10 21:12:17 -0500
49ba83fb (Jon Loeliger 2006-09-19 20:31:51 -0500
f8ff0c06 (Petr Baudis 2005-09-22 11:25:28 +0200
85023577 (Junio C Hamano 2006-12-19 14:34:12 -0800
85023577 (Junio C Hamano 2006-12-19 14:34:12 -0800
695dffe2 (Johannes Schindelin 2006-09-28 12:00:35 +0200
415e7b87 (Patrick Welche 2007-10-18 18:17:39 +0100
9048fe1c (Petr Baudis 2005-09-24 16:13:01 +0200
f8ff0c06 (Petr Baudis 2005-09-22 11:25:28 +0200
1955fabf (Mark Wooding 2006-02-03 20:27:04 +0000
f8ff0c06 (Petr Baudis 2005-09-22 11:25:28 +0200
960deccb (H. Peter Anvin 2005-10-19 14:27:01 -0700
1b1dd23f (Stephan Beyer 2008-07-13 15:36:15 +0200
3bd62c21 (Stephen R. van den Berg 2008-08-14 20:02:20 +0200
3bd62c21 (Stephen R. van den Berg 2008-08-14 20:02:20 +0200
73a7a656 (Jens Axboe 2007-07-27 14:00:29 -0700
49ba83fb (Jon Loeliger 2006-09-19 20:31:51 -0500
678dac6b (Tilman Sauerbeck 2006-08-22 19:37:41 +0200
d9edcbd6 (Junio C Hamano 2006-09-07 01:40:04 -0700
dd467629 (Jon Loeliger 2006-09-26 09:47:43 -0500
dd467629 (Jon Loeliger 2006-09-26 09:47:43 -0500
dd467629 (Jon Loeliger 2006-09-26 09:47:43 -0500
4ae95682 (H. Peter Anvin 2005-09-26 19:10:55 -0700
4ae95682 (H. Peter Anvin 2005-09-26 19:10:55 -0700
96f1e58f (David Rientjes
```

```
1) #include "cache.h"
2) #include "pkt-line.h"
3) #include "exec_cmd.h"
4) #include " interpolate.h"
5)
6) #include <syslog.h>
7)
8) #ifndef HOST_NAME_MAX
9) #define HOST_NAME_MAX 256
10) #endif
11)
12) #ifndef NI_MAXSERV
13) #define NI_MAXSERV 32
14) #endif
15)
16) static int log_syslog;
17) static int verbose;
18) static int reuseaddr;
19)
20) static const char daemon_usage[] =
21) "git daemon [--verbose] [--syslog] [
22) " --timeout=n ] [--init-ti
23) " --strict-paths] [--base
24) " --user-path | --user-pa
25) " --interpolated-path=pat
26) " --reuseaddr] [--detach]
27) " [--enable|disable|allow
28) " --inetd | [--listen=hos
29) " --user=user
30) " [directory...]";
31)
32) /* List of acceptable pathname prefi
33) static char **ok_paths;
```

git blame daemon.c

979e32fa (Randal L. Schwartz	2005-10-25 16:29:09 -0700	1) #include "cache.h"
85023577 (Junio C Hamano	2006-12-19 14:34:12 -0800	2) #include "pkt-line.h"
77cb17e9 (Michal Ostrowski	2006-01-10 21:12:17 -0500	3) #include "exec_cmd.h"
49ba83fb (Jon Loeliger	2006-09-19 20:31:51 -0500	4) #include " interpolate.h"
f8ff0c06 (Petr Baudis	2005-09-22 11:25:28 +0200	5)
85023577 (Junio C Hamano	2006-12-19 14:34:12 -0800	6) #include <syslog.h>
85023577 (Junio C Hamano	2006-12-19 14:34:12 -0800	7)
695dffe2 (Johannes Schindelin	2006-09-28 12:00:35 +0200	8) #ifndef HOST_NAME_MAX
695dffe2 (Johannes Schindelin	2006-09-28 12:00:35 +0200	9) #define HOST_NAME_MAX 256
695dffe2 (Johannes Schindelin	2006-09-28 12:00:35 +0200	10) #endif
695dffe2 (Johannes Schindelin	2006-09-28 12:00:35 +0200	11)
415e7b87 (Patrick Welche	2007-10-18 18:17:39 +0100	12) #ifndef NI_MAXSERV
415e7b87 (Patrick Welche	2007-10-18 18:17:39 +0100	13) #define NI_MAXSERV 32
415e7b87 (Patrick Welche	2007-10-18 18:17:39 +0100	14) #endif
415e7b87 (Patrick Welche	2007-10-18 18:17:39 +0100	15)
9048fe1c (Petr Baudis	2005-09-24 16:13:01 +0200	16) static int log_syslog;
f8ff0c06 (Petr Baudis	2005-09-22 11:25:28 +0200	17) static int verbose;
1955fabf (Mark Wooding	2006-02-03 20:27:04 +0000	18) static int reuseaddr;
f8ff0c06 (Petr Baudis	2005-09-22 11:25:28 +0200	19)
960deccb (H. Peter Anvin	2005-10-19 14:27:01 -0700	20) static const char daemon_usage[] =
1b1dd23f (Stephan Beyer	2008-07-13 15:36:15 +0200	21) "git daemon [--verbose] [--syslog] [
3bd62c21 (Stephen R. van den Berg	2008-08-14 20:02:20 +0200	22) " --timeout=n] [--init-ti
3bd62c21 (Stephen R. van den Berg	2008-08-14 20:02:20 +0200	23) " --strict-paths] [--base
73a7a656 (Jens Axboe	2007-07-27 14:00:29 -0700	24) " --user-path --user-pa
49ba83fb (Jon Loeliger	2006-09-19 20:31:51 -0500	25) " --interpolated-path=pat
678dac6b (Tilman Sauerbeck	2006-08-22 19:37:41 +0200	26) " --reuseaddr] [--detach]
d9edcbd6 (Junio C Hamano	2006-09-07 01:40:04 -0700	27) " --[enable disable allow
dd467629 (Jon Loeliger	2006-09-26 09:47:43 -0500	28) " --inetd [--listen=hos
dd467629 (Jon Loeliger	2006-09-26 09:47:43 -0500	29) " [--user=user
dd467629 (Jon Loeliger	2006-09-26 09:47:43 -0500	30) " [directory...]";
4ae95682 (H. Peter Anvin	2005-09-26 19:10:55 -0700	31)
4ae95682 (H. Peter Anvin	2005-09-26 19:10:55 -0700	32) /* List of acceptable pathname prefi
96f1e58f (David Rienties	2006-08-15 10:23:48 -0700	33) static char **ok_paths:

```
git blame -c GITPackUpload.m
```

git blame -C GITPackUpload.m

```
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 12)
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 13) #define PACK_SIGN
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 14) #define PACK_VERS
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 15)
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 16) @implementation G
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 17)
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 18) @synthesize gitRe
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 19) @synthesize needR
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 20) @synthesize gitSo
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 21) @synthesize refDi
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 22)
a2cbabf5 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-25 22:29:39 +0100 23) - (id) initWithGi
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 24) {
a2cbabf5 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-25 22:29:39 +0100 25)     gitRepo =
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 26)     needRefs =
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 27)     gitSocket
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 28)     return sel
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 29) }
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 30)
ad11ac80 Source/Network/GITPackUpload.m (Scott Chacon 2009-03-24 18:32:50 +0100 31) - (bool) uploadPa
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 32) {
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 33)     NSLog(@"up
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 34)     NSString *
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 35)     NSArray *t
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 36)
56ef2caf Source/Network/GITServerHandler.m (Scott Chacon 2009-01-05 21:44:26 -0800 37)     refDict =
f344f58d Source/Network/GITServerHandler.m (Scott Chacon 2009-01-04 18:59:04 -0800 38)
```

git blame -C GITPackUpload.m

f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 12)
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 13) #define PACK_SIGN
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 14) #define PACK_VERS
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 15)
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 16) @implementation G
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 17)
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 18) @synthesize gitRe
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 19) @synthesize needR
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 20) @synthesize gitSo
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 21) @synthesize refDi
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 22)
a2cbabf5	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-25 22:29:39 +0100 23) - (id) initWithGi
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 24) {
a2cbabf5	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-25 22:29:39 +0100 25) gitRepo =
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 26) needRefs =
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 27) gitSocket
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 28) return sel
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 29) }
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 30)
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 31) - (bool) uploadPa
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 32) {
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 33) NSLog(@"up
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 34) NSString *
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 35) NSArray *t
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 36)
56ef2caf	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-05 21:44:26 -0800 37) refDict =
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 38)

git blame -C GITPackUpload.m

f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 12)	
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 13) #define PACK_SIGN	
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 14) #define PACK_VERS	
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 15)	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 16) @implementation G	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 17)	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 18) @synthesize gitRe	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 19) @synthesize needR	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 20) @synthesize gitSo	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 21) @synthesize refDi	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 22)	
a2cbabf5	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-25 22:29:39 +0100 23) - (id) initWithGi	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 24) {	
a2cbabf5	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-25 22:29:39 +0100 25) gitRepo =	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 26) needRefs =	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 27) gitSocket	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 28) return sel	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 29) }	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 30)	
ad11ac80	Source/Network/GITPackUpload.m	(Scott Chacon 2009-03-24 18:32:50 +0100 31) - (bool) uploadPa	
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 32) {	
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 33) NSLog(@"up	
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 34) NSString *	
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 35) NSArray *t	
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 36)	
56ef2caf	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-05 21:44:26 -0800 37) refDict =	
f344f58d	Source/Network/GITServerHandler.m	(Scott Chacon 2009-01-04 18:59:04 -0800 38)	

Bisecting

**binary search for where
a bug was introduced**

```
$ git bisect start
```

```
$ git bisect start  
$ git bisect bad
```

```
$ git bisect start  
$ git bisect bad  
$ git bisect good 3acb4c2c6666ed6cb91cb0b983efe9d50cf8cfbe
```

```
$ git bisect start
$ git bisect bad
$ git bisect good 3acb4c2c6666ed6cb91cb0b983efe9d50cf8cfbe
Bisecting: 6 revisions left to test after this
[ecb6e1bc347ccecc5f9350d878ce677feb13d3b2] error handling on repo
```

```
$ git bisect start
$ git bisect bad
$ git bisect good 3acb4c2c6666ed6cb91cb0b983efe9d50cf8cfbe
Bisecting: 6 revisions left to test after this
[ecb6e1bc347ccecc5f9350d878ce677feb13d3b2] error handling on repo
$ git bisect good
```

```
$ git bisect start
$ git bisect bad
$ git bisect good 3acb4c2c6666ed6cb91cb0b983efe9d50cf8cfbe
Bisecting: 6 revisions left to test after this
[ecb6e1bc347ccecc5f9350d878ce677feb13d3b2] error handling on repo
$ git bisect good
Bisecting: 3 revisions left to test after this
[b047b02ea83310a70fd603dc8cd7a6cd13d15c04] secure this thing
```

```
$ git bisect start
$ git bisect bad
$ git bisect good 3acb4c2c6666ed6cb91cb0b983efe9d50cf8cfbe
Bisecting: 6 revisions left to test after this
[ecb6e1bc347ccecc5f9350d878ce677feb13d3b2] error handling on repo
$ git bisect good
Bisecting: 3 revisions left to test after this
[b047b02ea83310a70fd603dc8cd7a6cd13d15c04] secure this thing
$ git bisect bad
```

```
$ git bisect start
$ git bisect bad
$ git bisect good 3acb4c2c6666ed6cb91cb0b983efe9d50cf8cfbe
Bisecting: 6 revisions left to test after this
[ecb6e1bc347ccecc5f9350d878ce677feb13d3b2] error handling on repo
$ git bisect good
Bisecting: 3 revisions left to test after this
[b047b02ea83310a70fd603dc8cd7a6cd13d15c04] secure this thing
$ git bisect bad
Bisecting: 1 revisions left to test after this
[f71ce38690acf49c1f3c9bea38e09d82a5ce6014] drop exceptions table
```

```
$ git bisect start
$ git bisect bad
$ git bisect good 3acb4c2c6666ed6cb91cb0b983efe9d50cf8cfbe
Bisecting: 6 revisions left to test after this
[ecb6e1bc347ccecc5f9350d878ce677feb13d3b2] error handling on repo
$ git bisect good
Bisecting: 3 revisions left to test after this
[b047b02ea83310a70fd603dc8cd7a6cd13d15c04] secure this thing
$ git bisect bad
Bisecting: 1 revisions left to test after this
[f71ce38690acf49c1f3c9bea38e09d82a5ce6014] drop exceptions table
$ git bisect good
```

```
$ git bisect start
$ git bisect bad
$ git bisect good 3acb4c2c6666ed6cb91cb0b983efe9d50cf8cfbe
Bisecting: 6 revisions left to test after this
[ecb6e1bc347ccecc5f9350d878ce677feb13d3b2] error handling on repo
$ git bisect good
Bisecting: 3 revisions left to test after this
[b047b02ea83310a70fd603dc8cd7a6cd13d15c04] secure this thing
$ git bisect bad
Bisecting: 1 revisions left to test after this
[f71ce38690acf49c1f3c9bea38e09d82a5ce6014] drop exceptions table
$ git bisect good
b047b02ea83310a70fd603dc8cd7a6cd13d15c04 is first bad commit
commit b047b02ea83310a70fd603dc8cd7a6cd13d15c04
Author: PJ Hyett <pjhyett@gmail.com>
Date: Tue Jan 27 14:48:32 2009 -0800
```

secure this thing

```
:040000 040000 40ee3e7821b895e52c1695092db9bdc4c61d1730
f24d3c6ebcf639b1a3814550e62d60b8e68a8e4 M config
```

```
$ git bisect start
$ git bisect bad
$ git bisect good 3acb4c2c6666ed6cb91cb0b983efe9d50cf8cfbe
Bisecting: 6 revisions left to test after this
[ecb6e1bc347ccecc5f9350d878ce677feb13d3b2] error handling on repo
$ git bisect good
Bisecting: 3 revisions left to test after this
[b047b02ea83310a70fd603dc8cd7a6cd13d15c04] secure this thing
$ git bisect bad
Bisecting: 1 revisions left to test after this
[f71ce38690acf49c1f3c9bea38e09d82a5ce6014] drop exceptions table
$ git bisect good
b047b02ea83310a70fd603dc8cd7a6cd13d15c04 is first bad commit
commit b047b02ea83310a70fd603dc8cd7a6cd13d15c04
Author: PJ Hyett <pjhyett@gmail.com>
Date: Tue Jan 27 14:48:32 2009 -0800
```

secure this thing

```
:040000 040000 40ee3e7821b895e52c1695092db9bdc4c61d1730
f24d3c6ebcf639b1a3814550e62d60b8e68a8e4 M config
```

```
$ git bisect reset
```

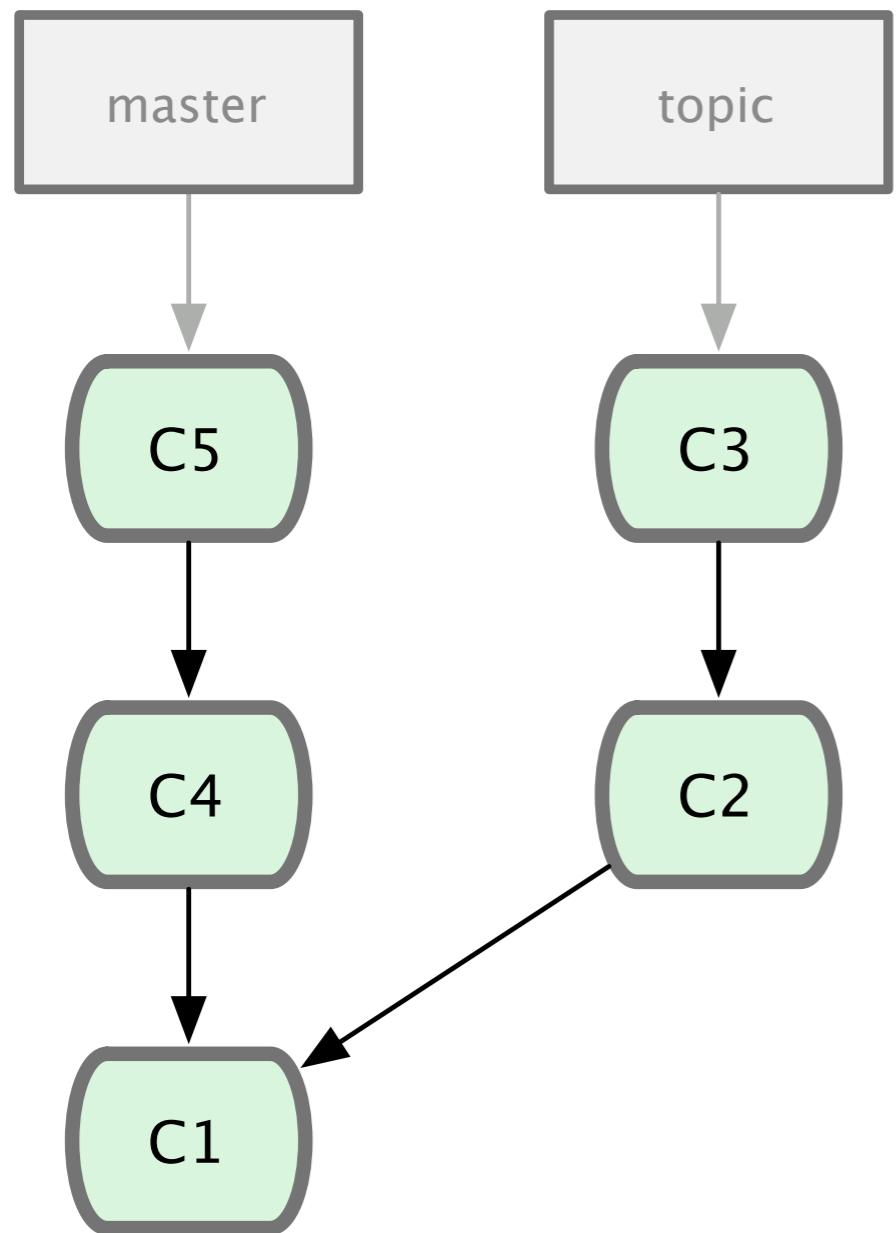
Rebasing

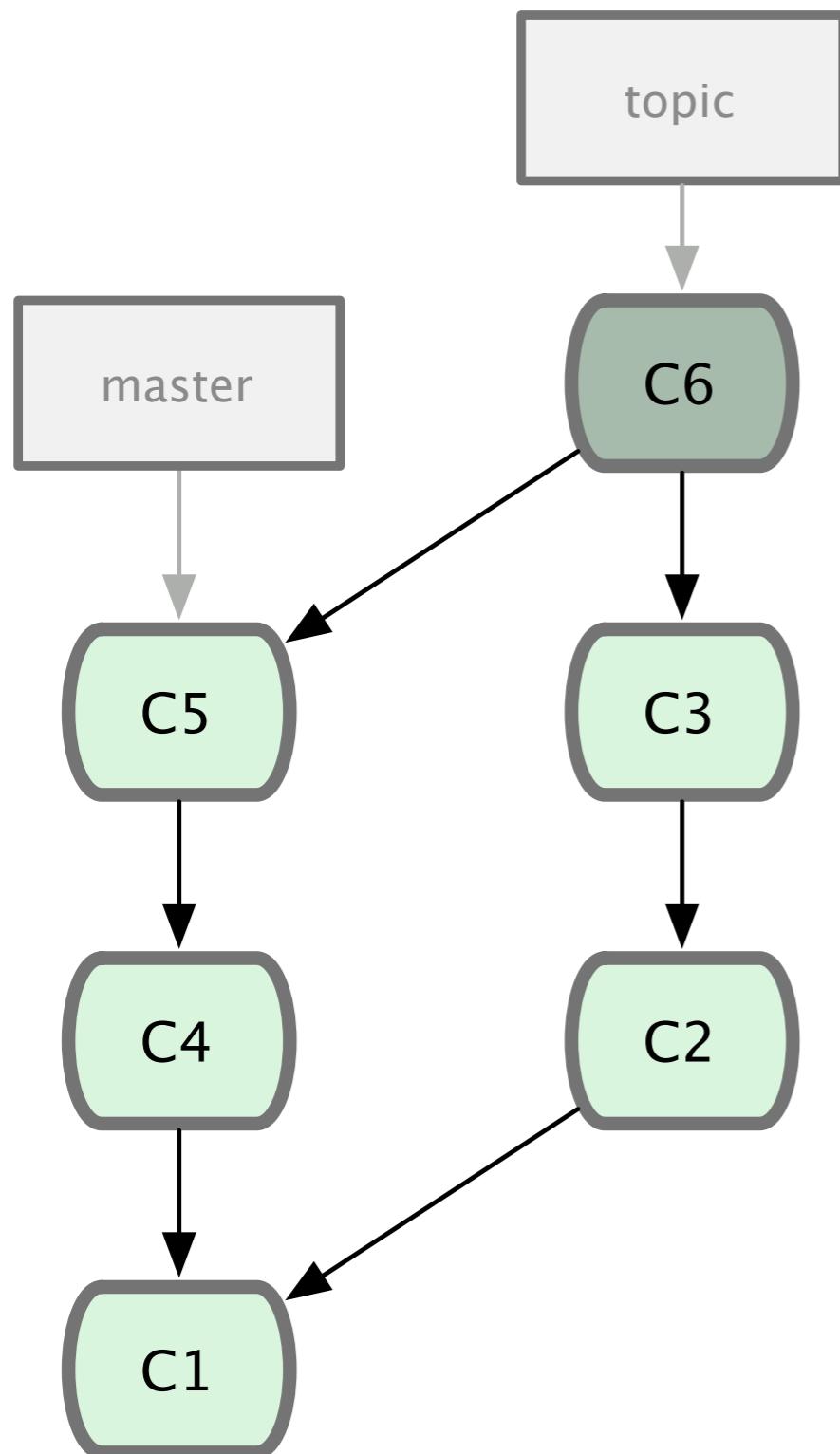
Rewriting History

Modifying the last commit

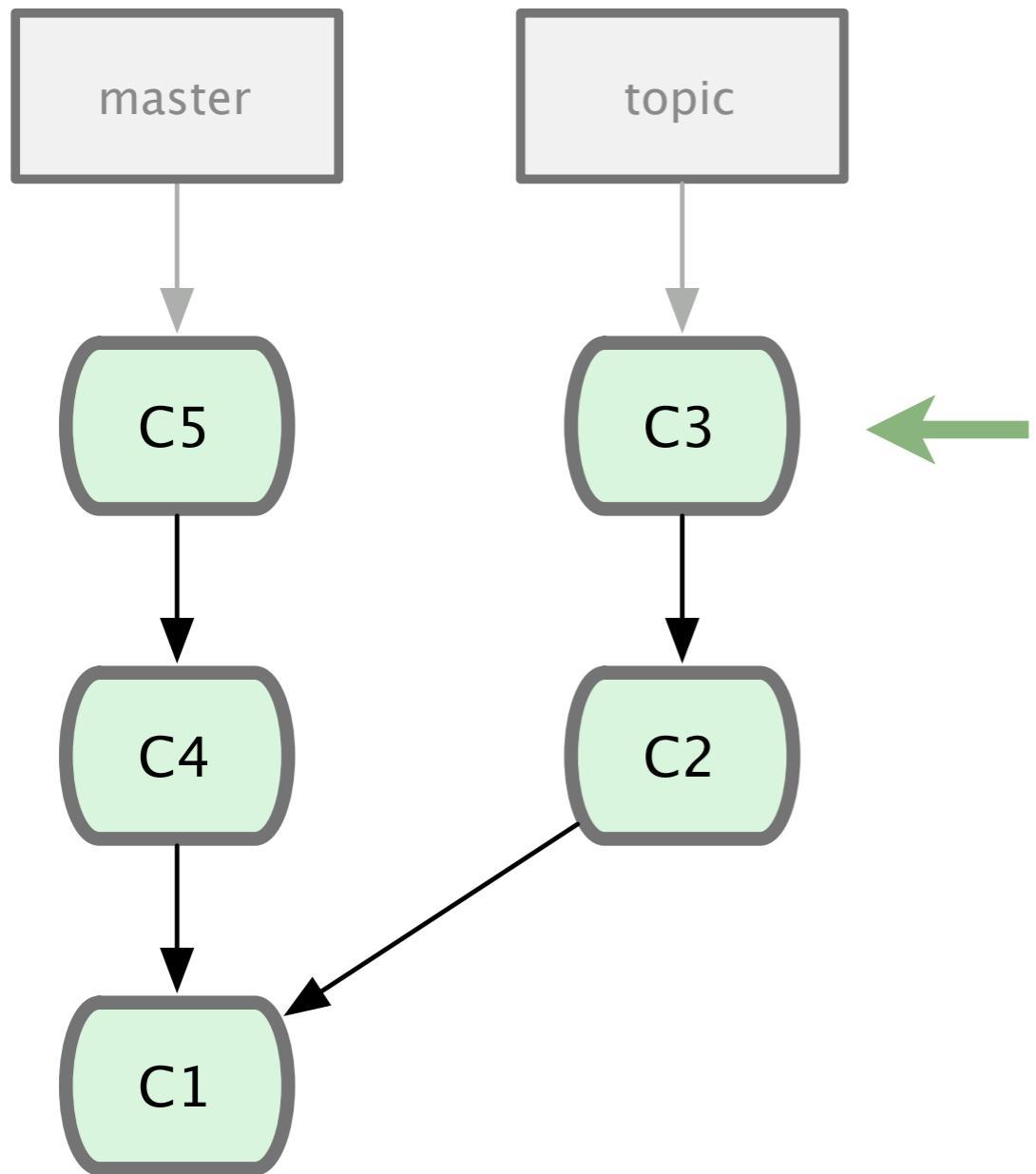
git commit --amend

Rebasing

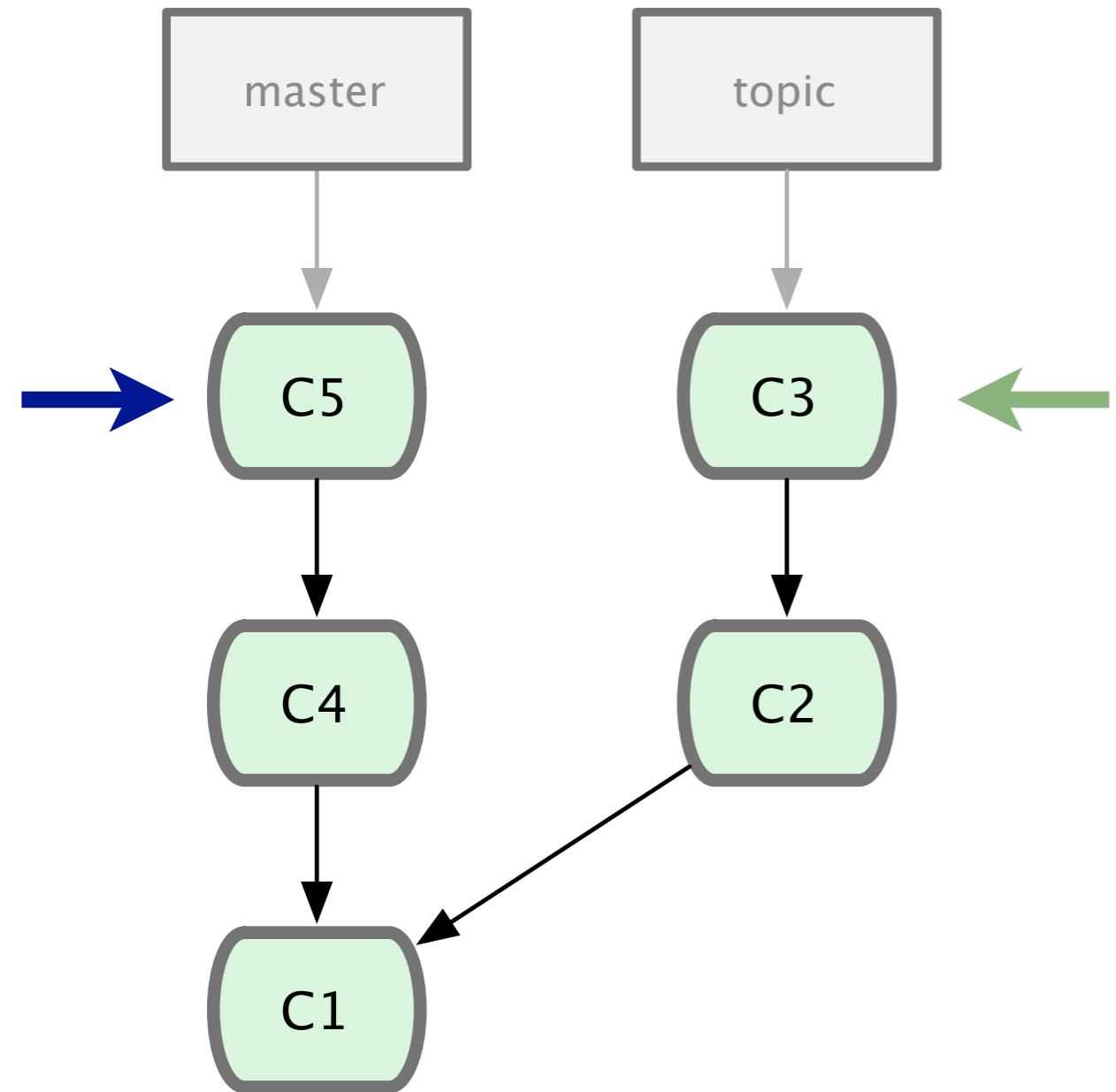




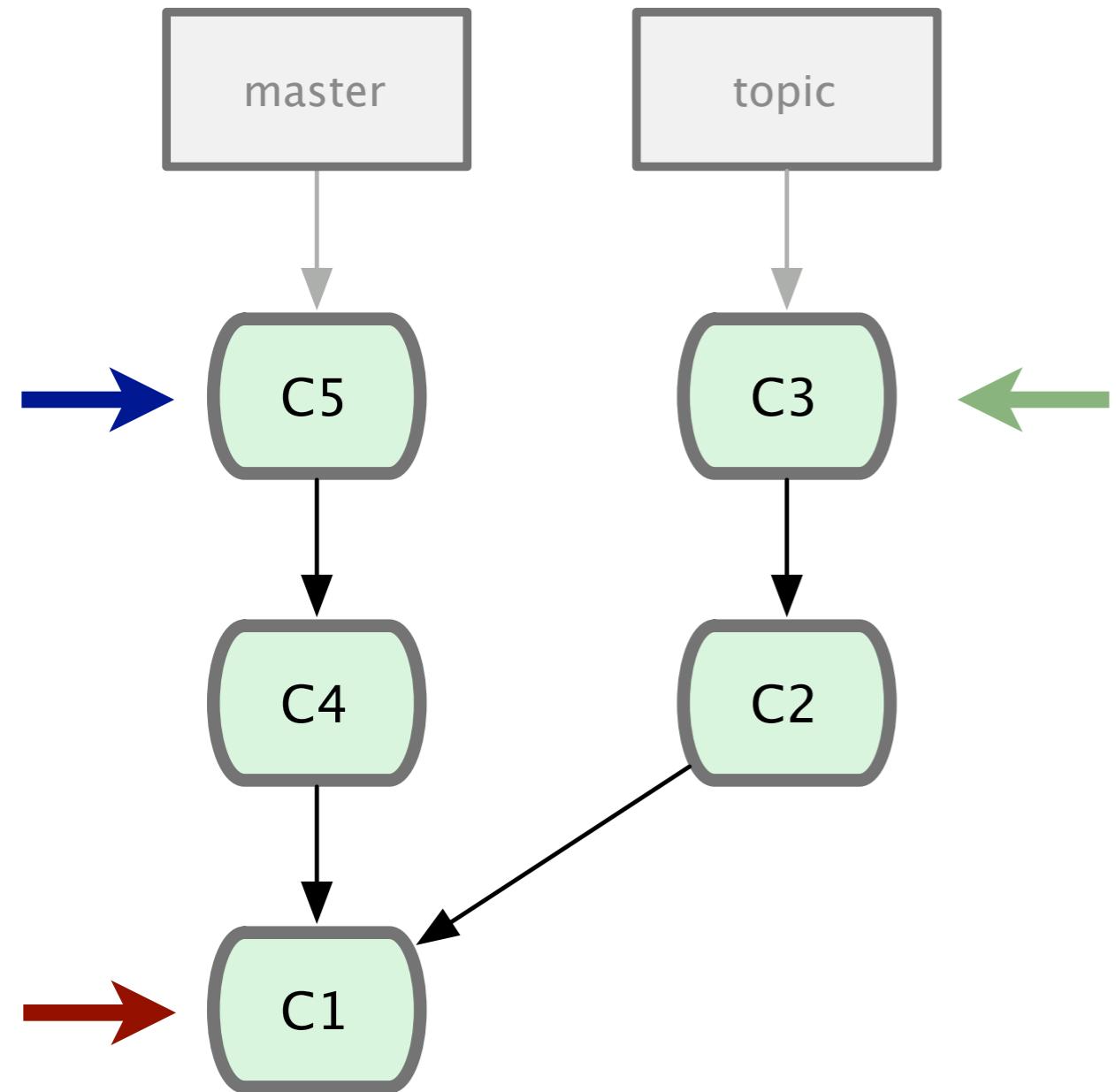
git merge master



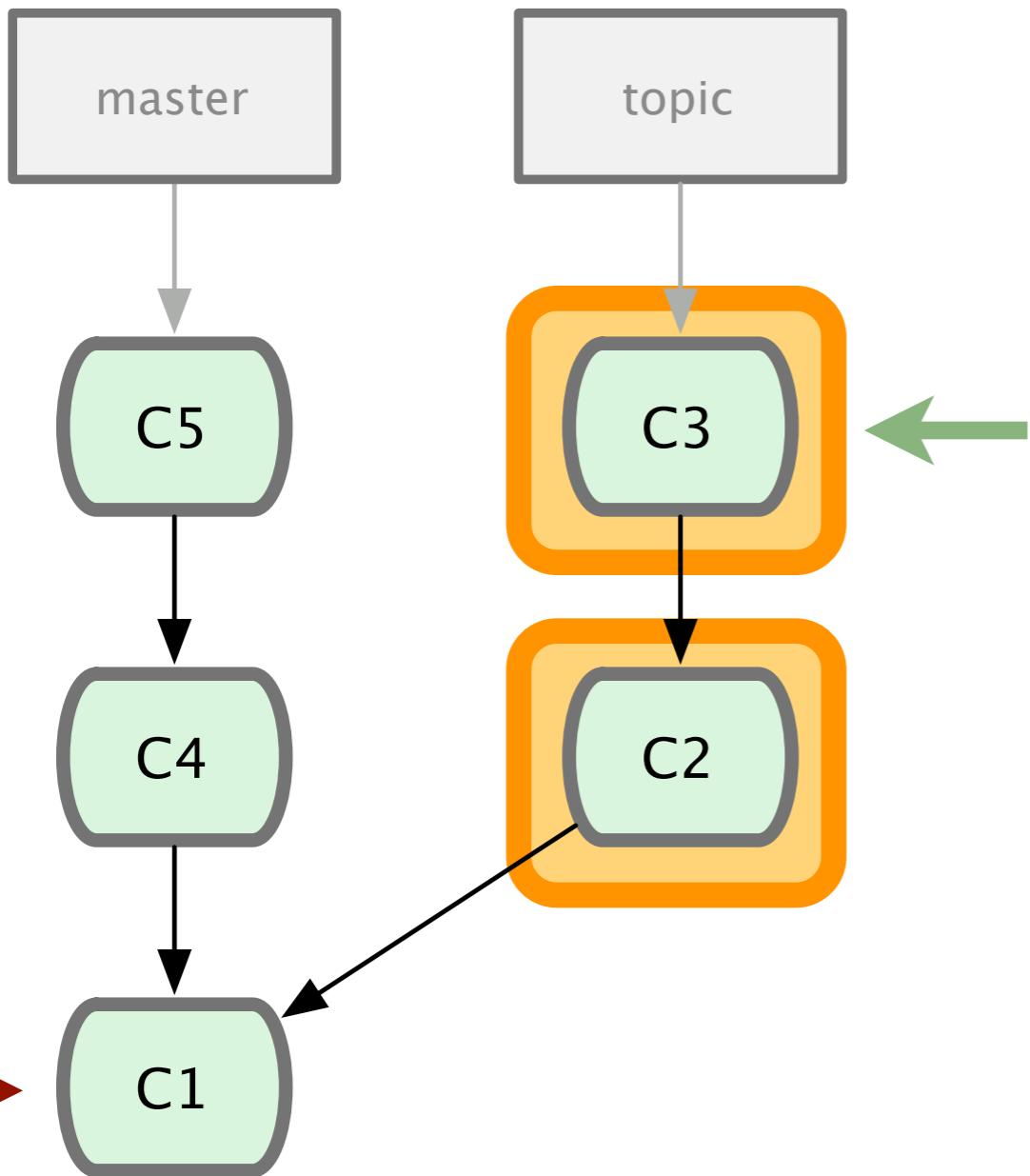
git rebase master



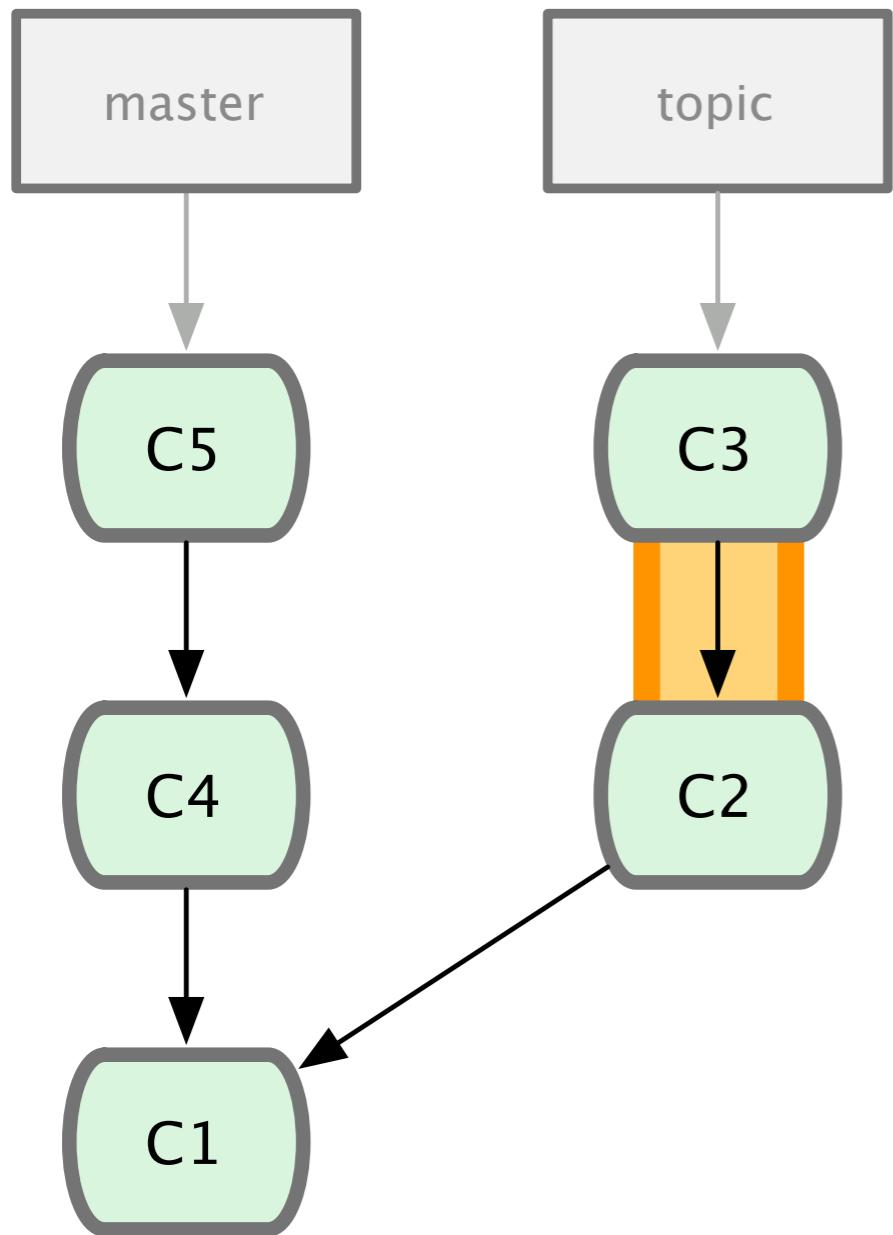
git rebase master



git rebase master

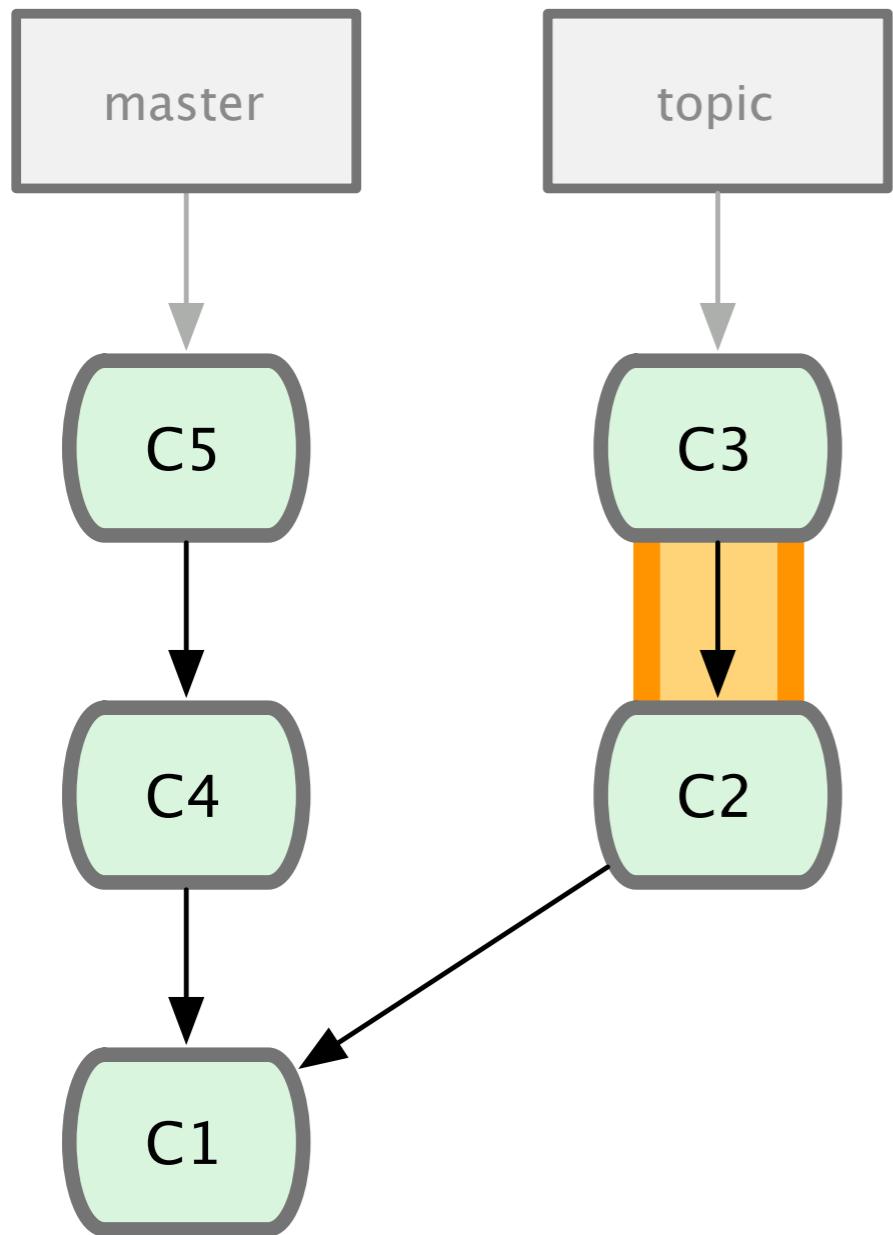


git rebase master



`git diff c2 c3 > 2-3.patch`

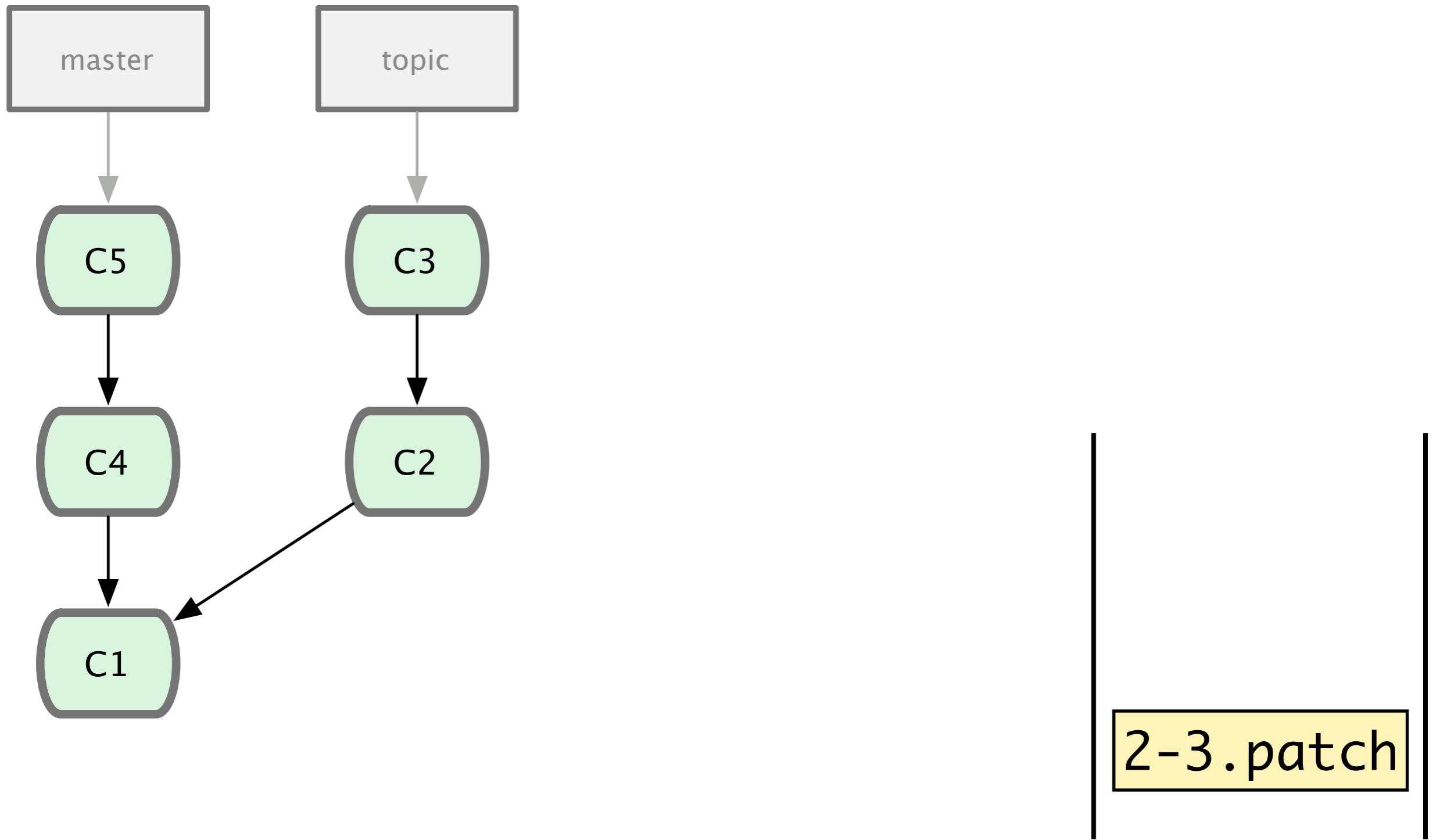
`git rebase master`

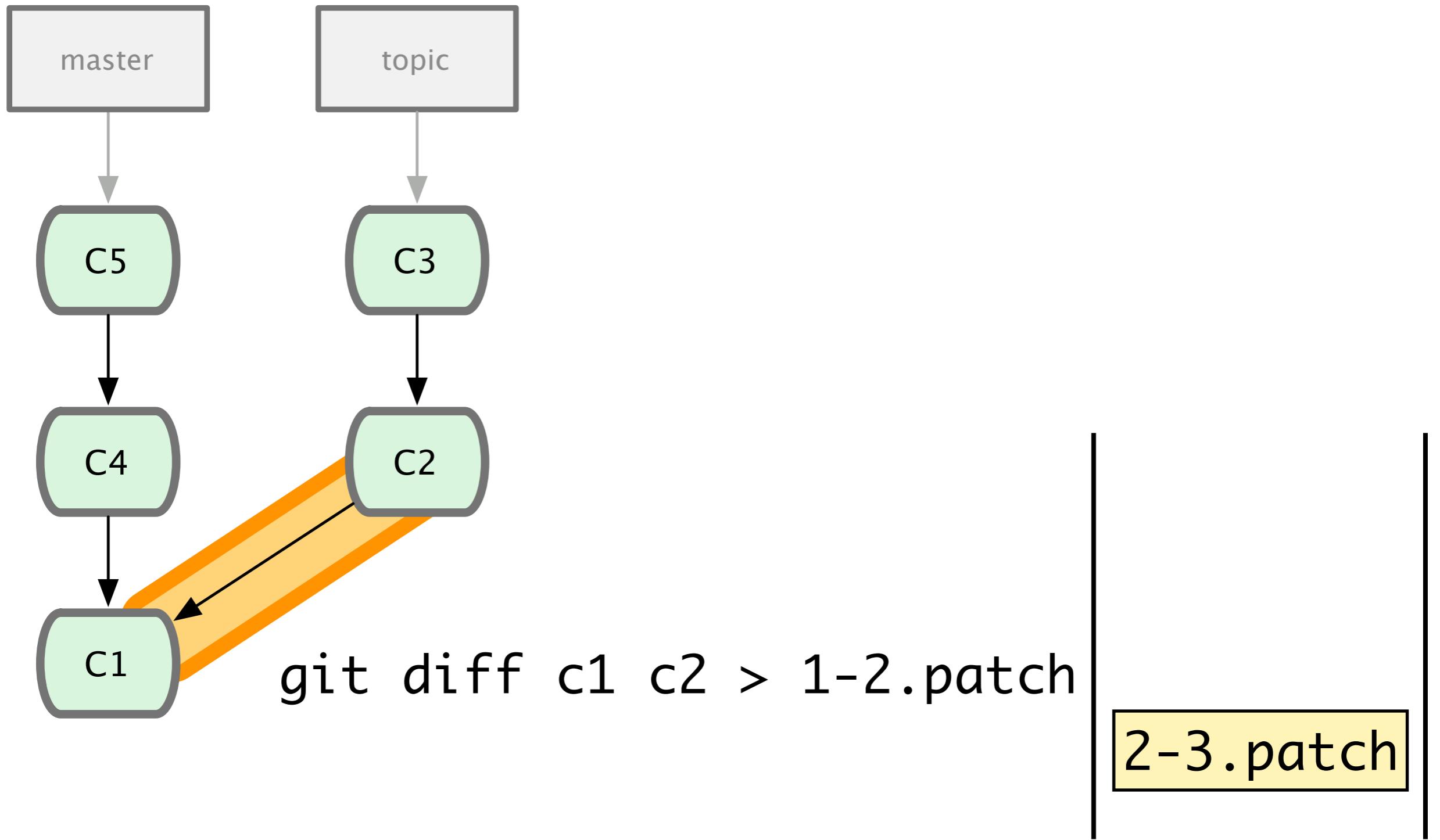


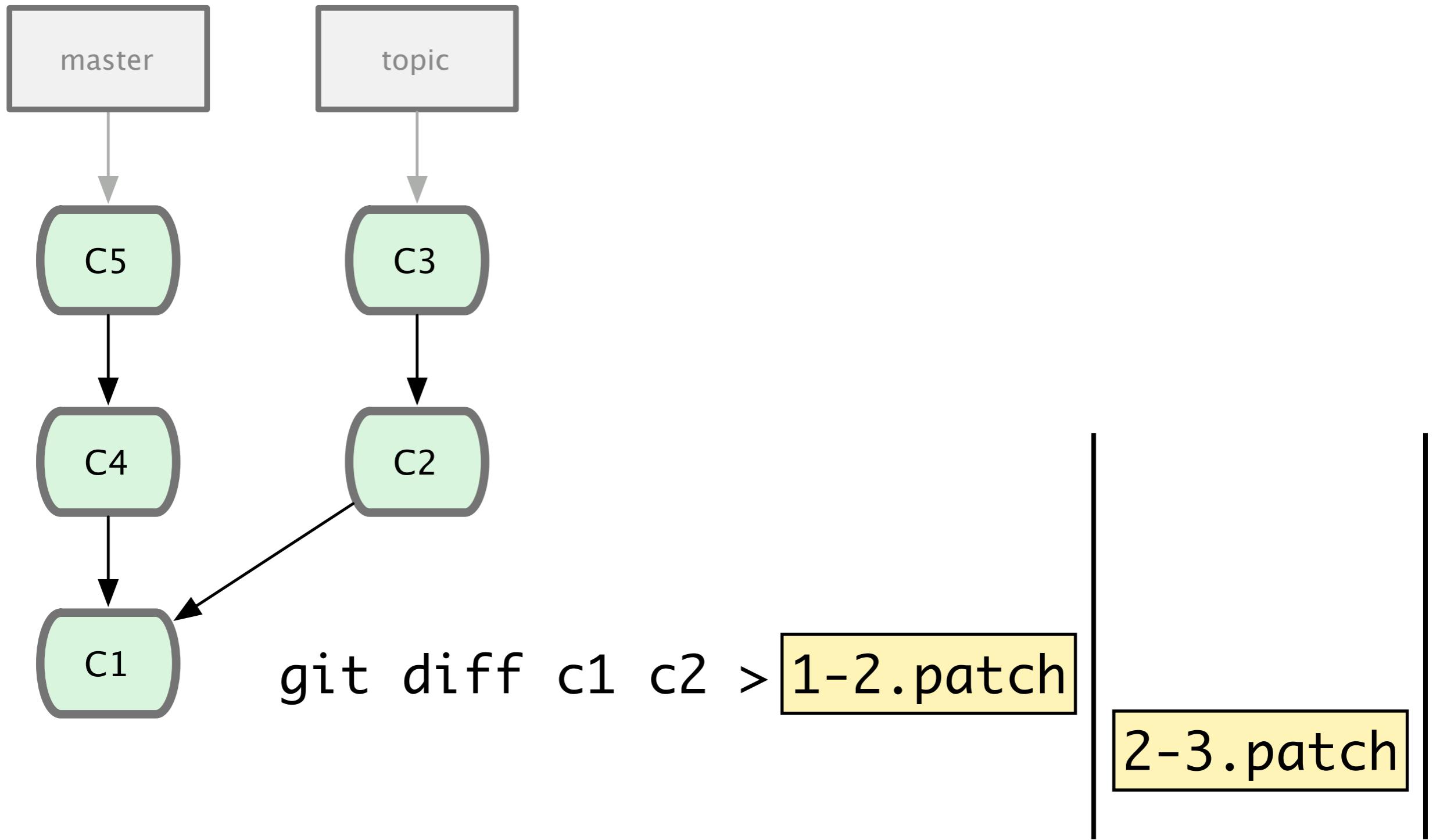
git diff c2 c3 > 2-3.patch

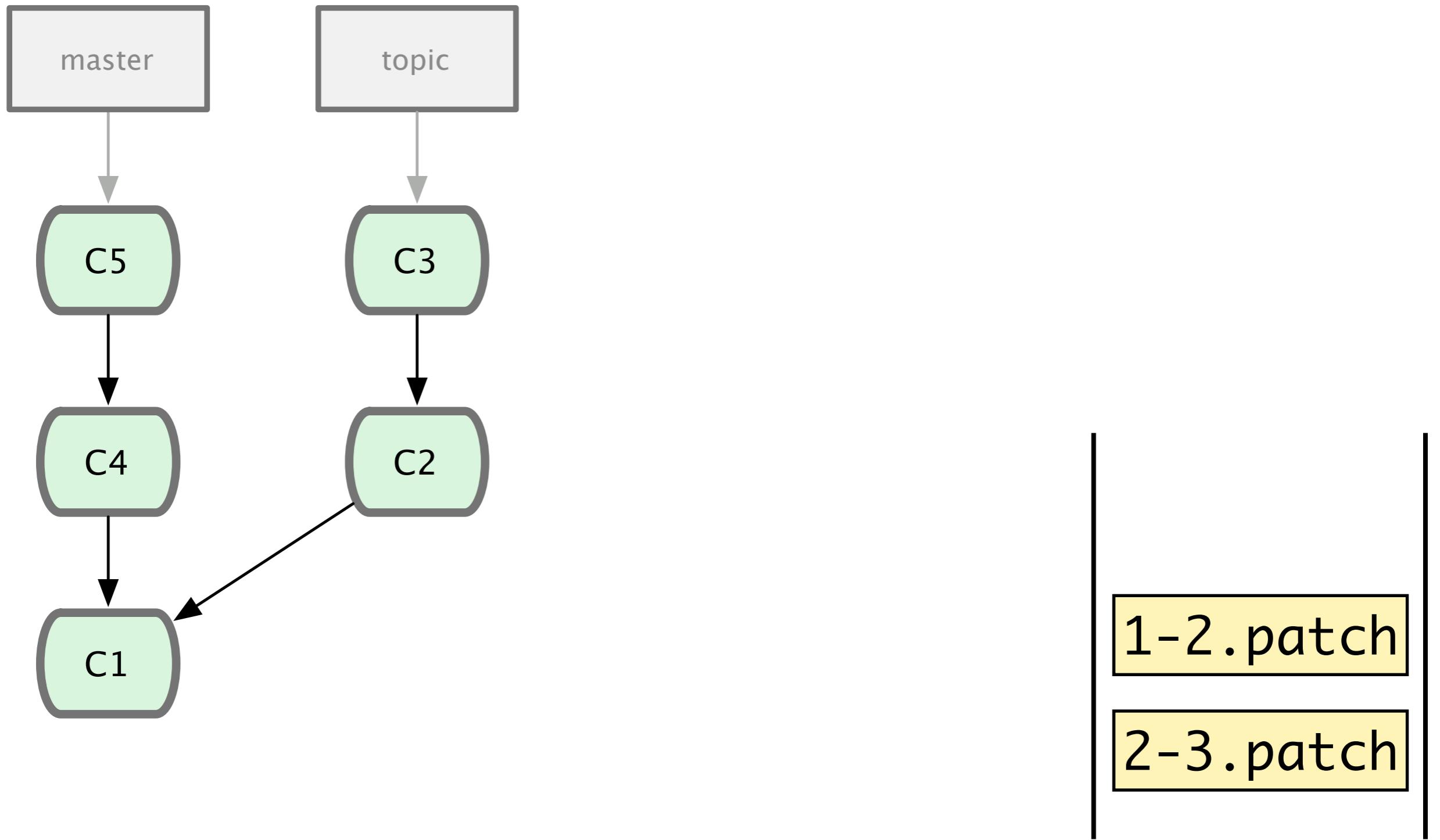
```
diff --git a/test b/test
index 2eadcec..bd8c6c9 100644
--- a/test
+++ b/test
@@ -1,2 +1,3 @@
version one
version four
+version five
```

git rebase master

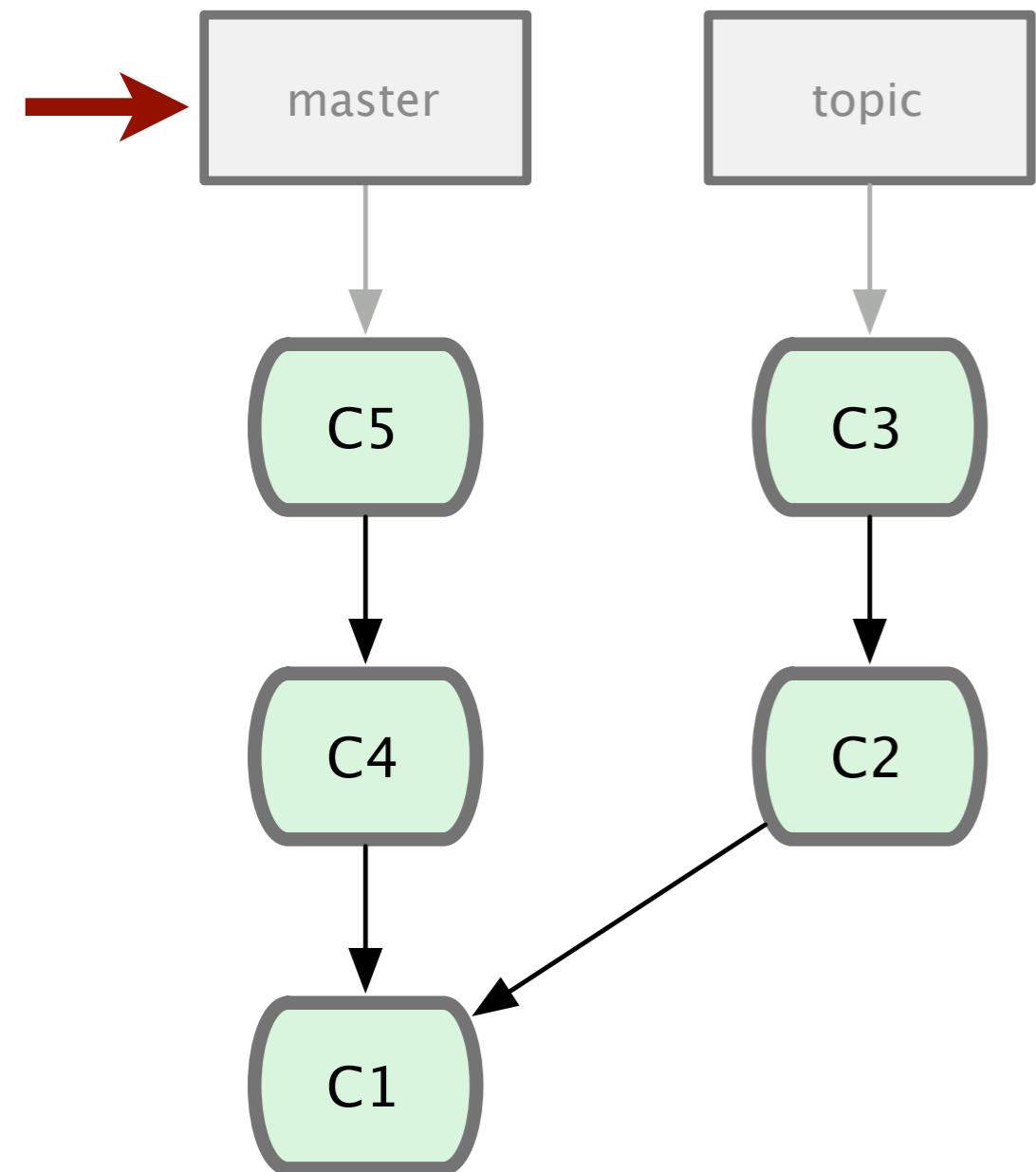








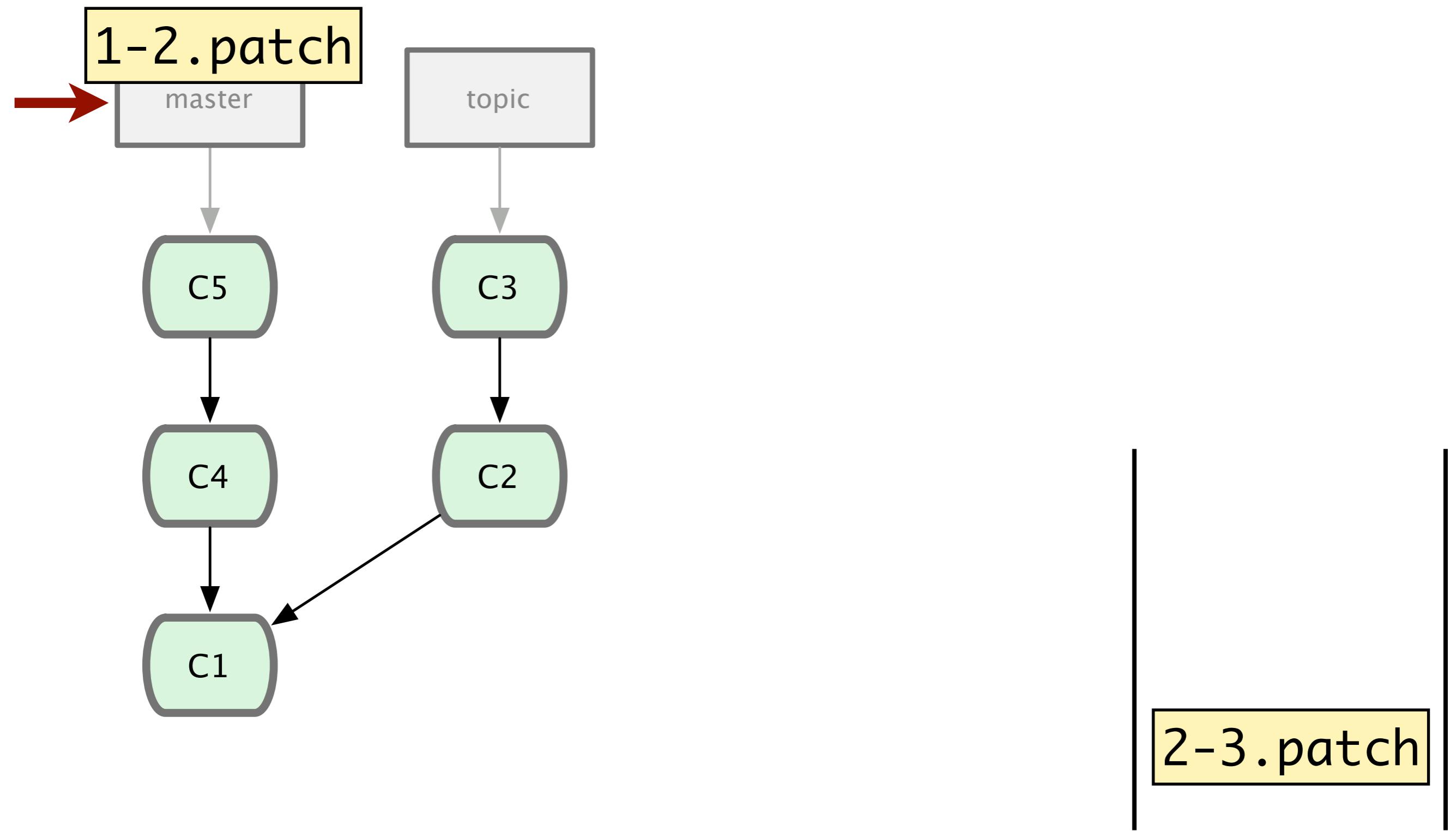
git rebase master



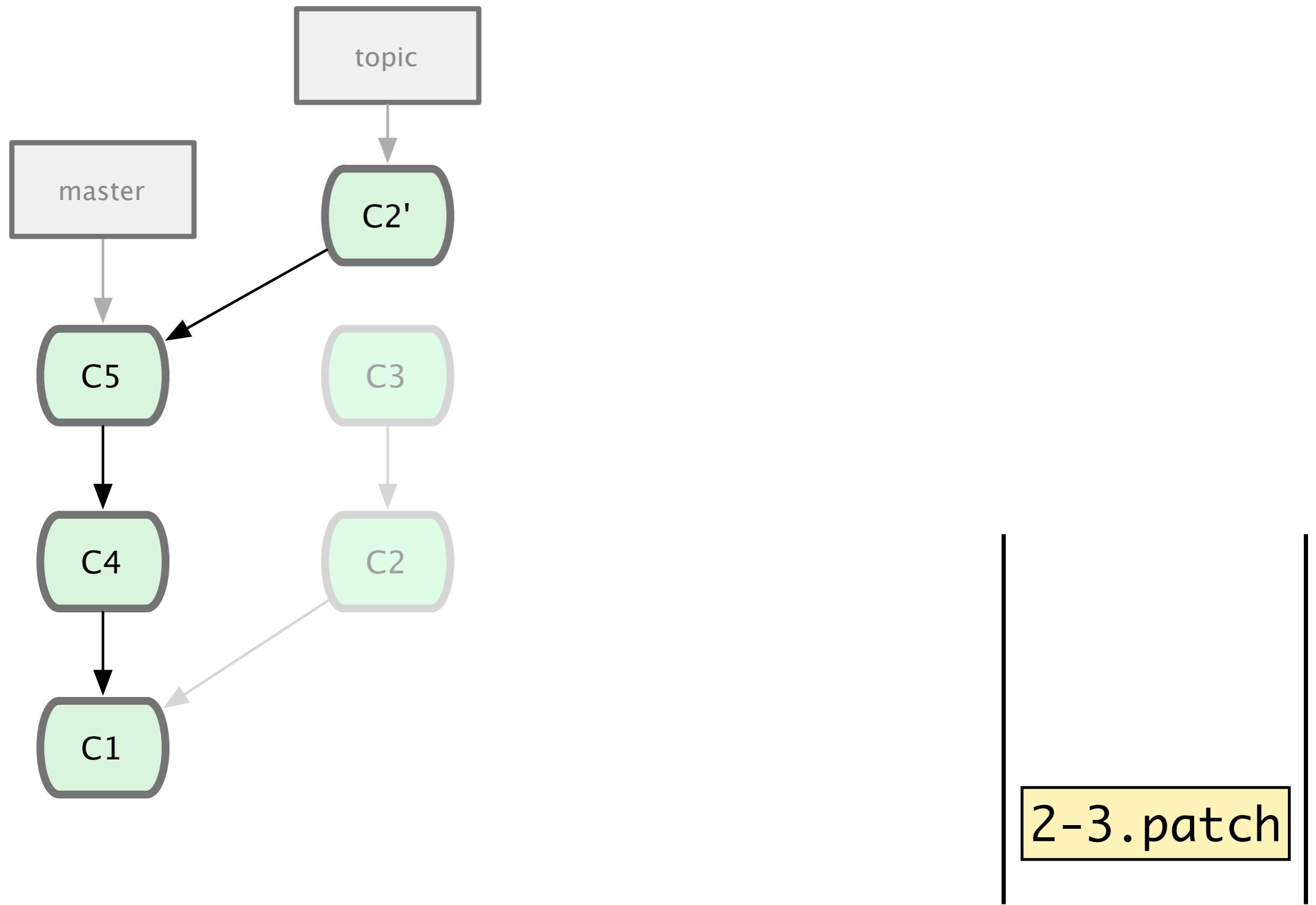
1-2.patch

2-3.patch

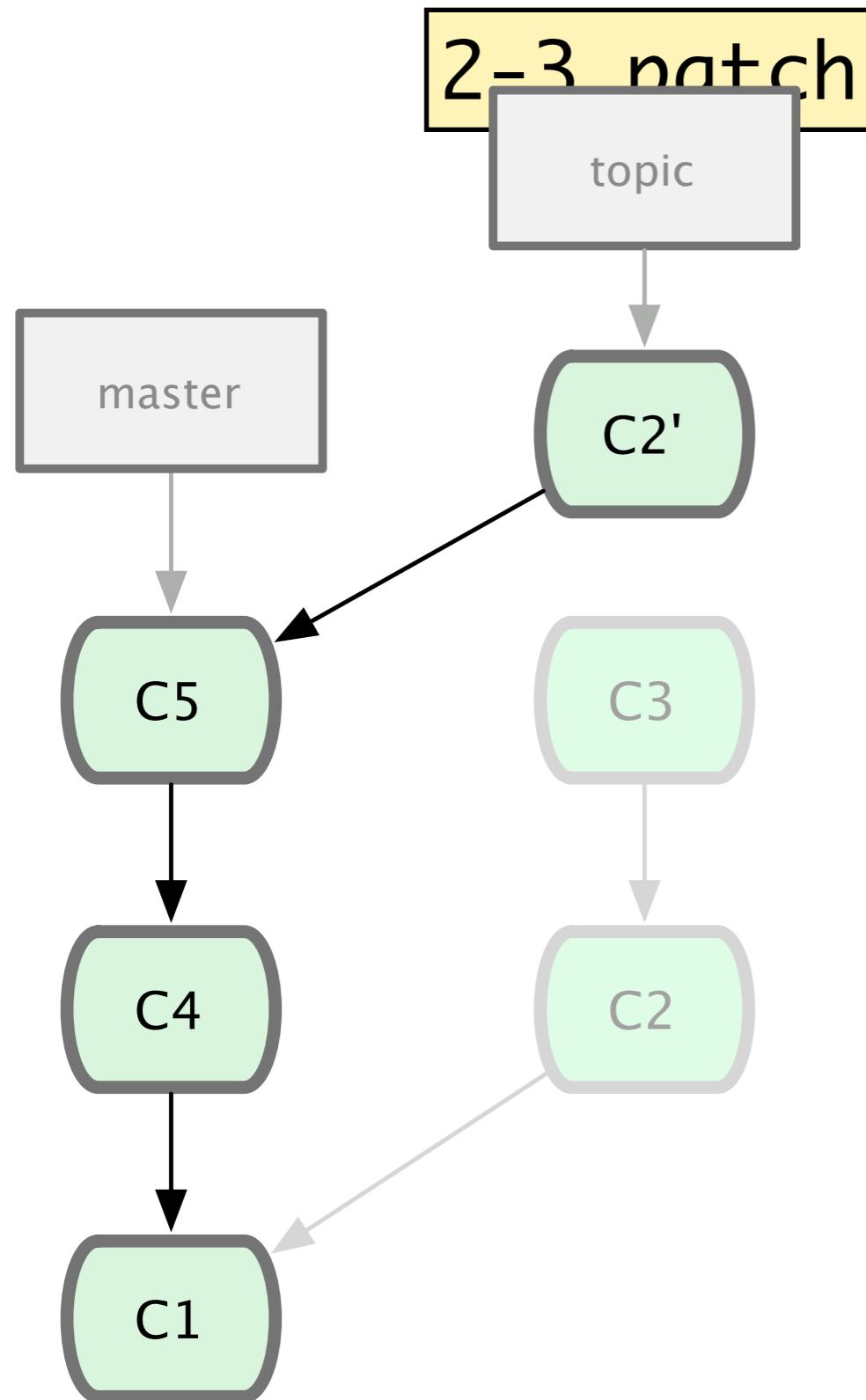
git rebase master

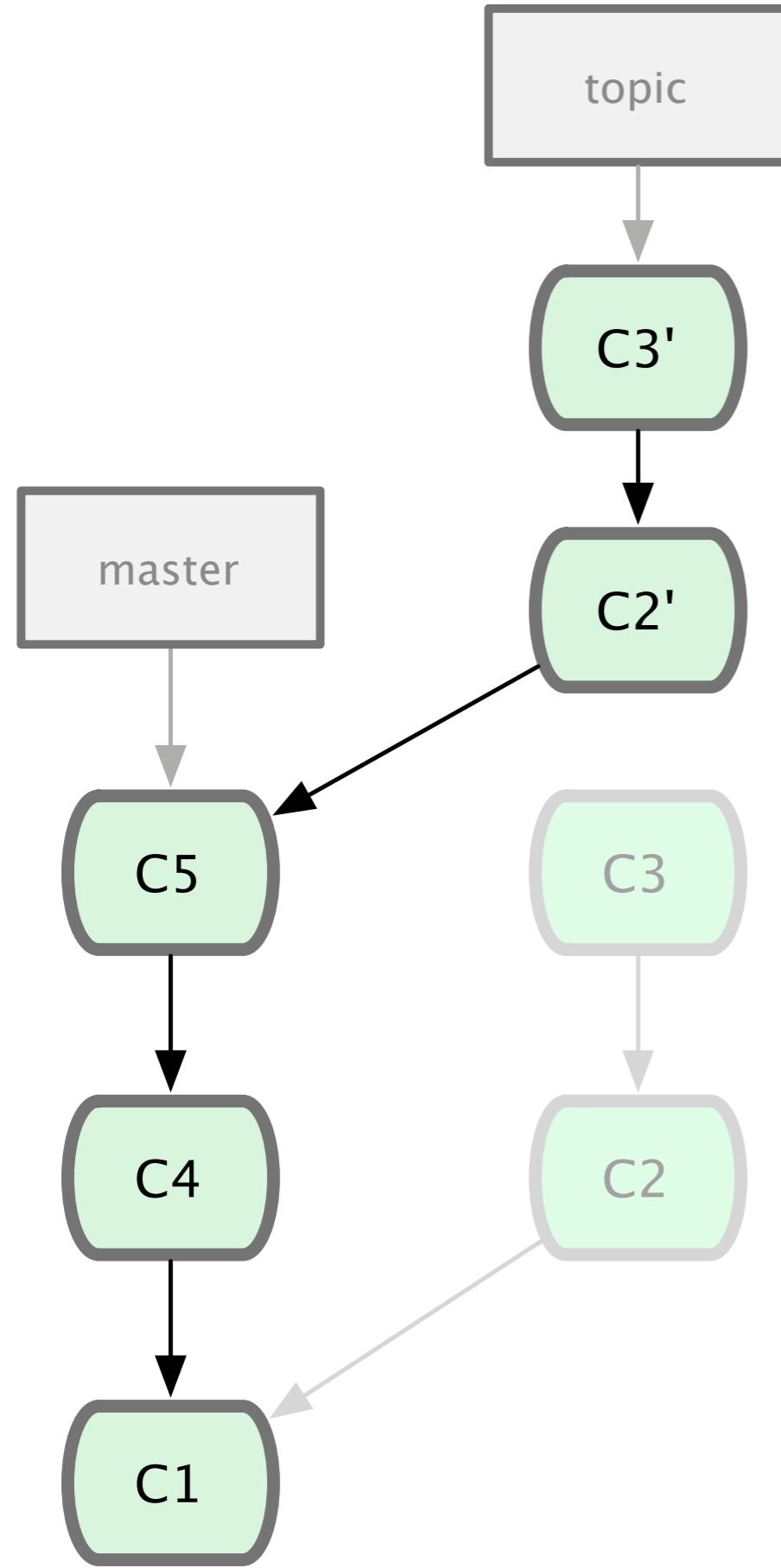


git rebase master

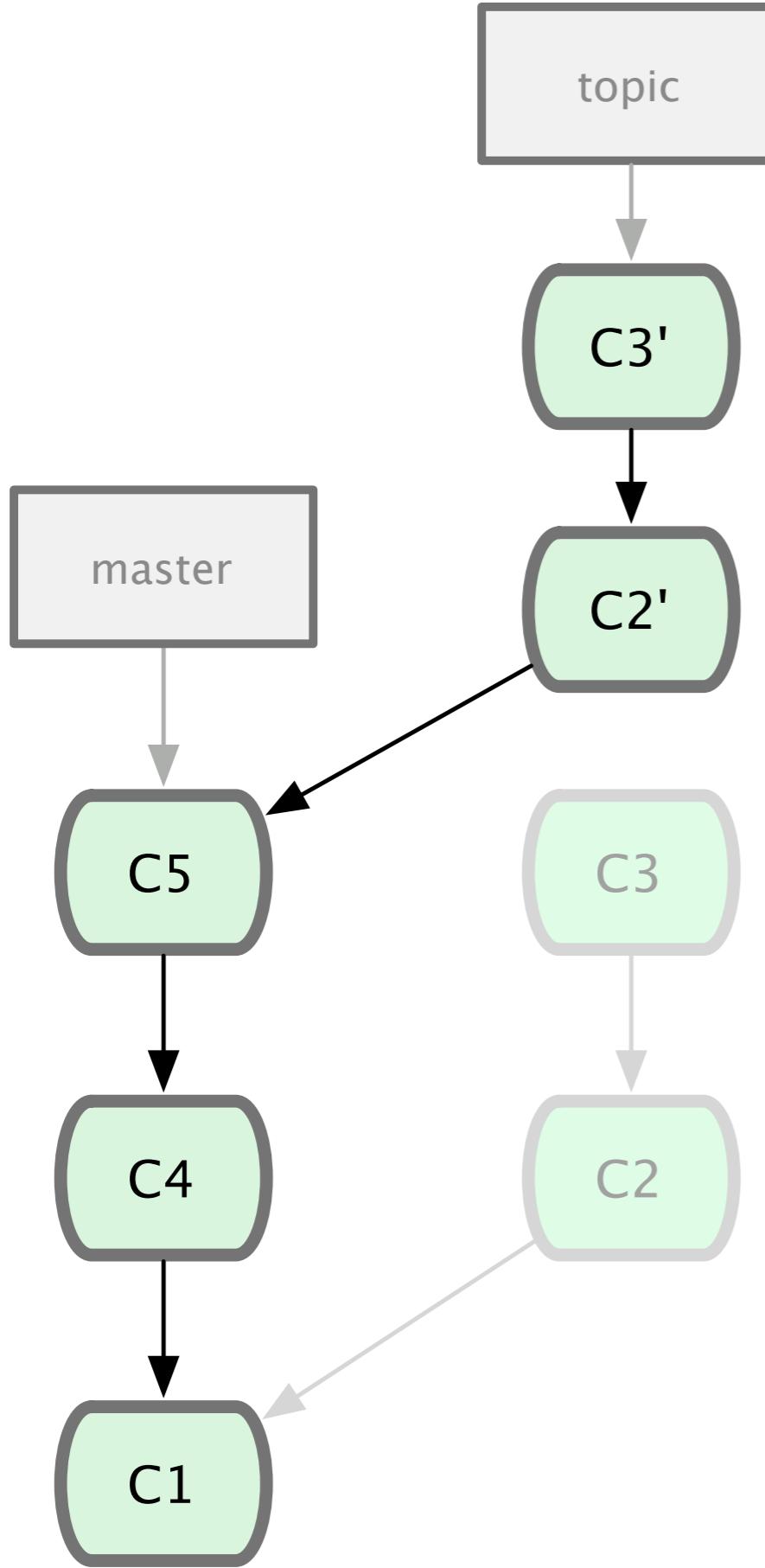


git rebase master

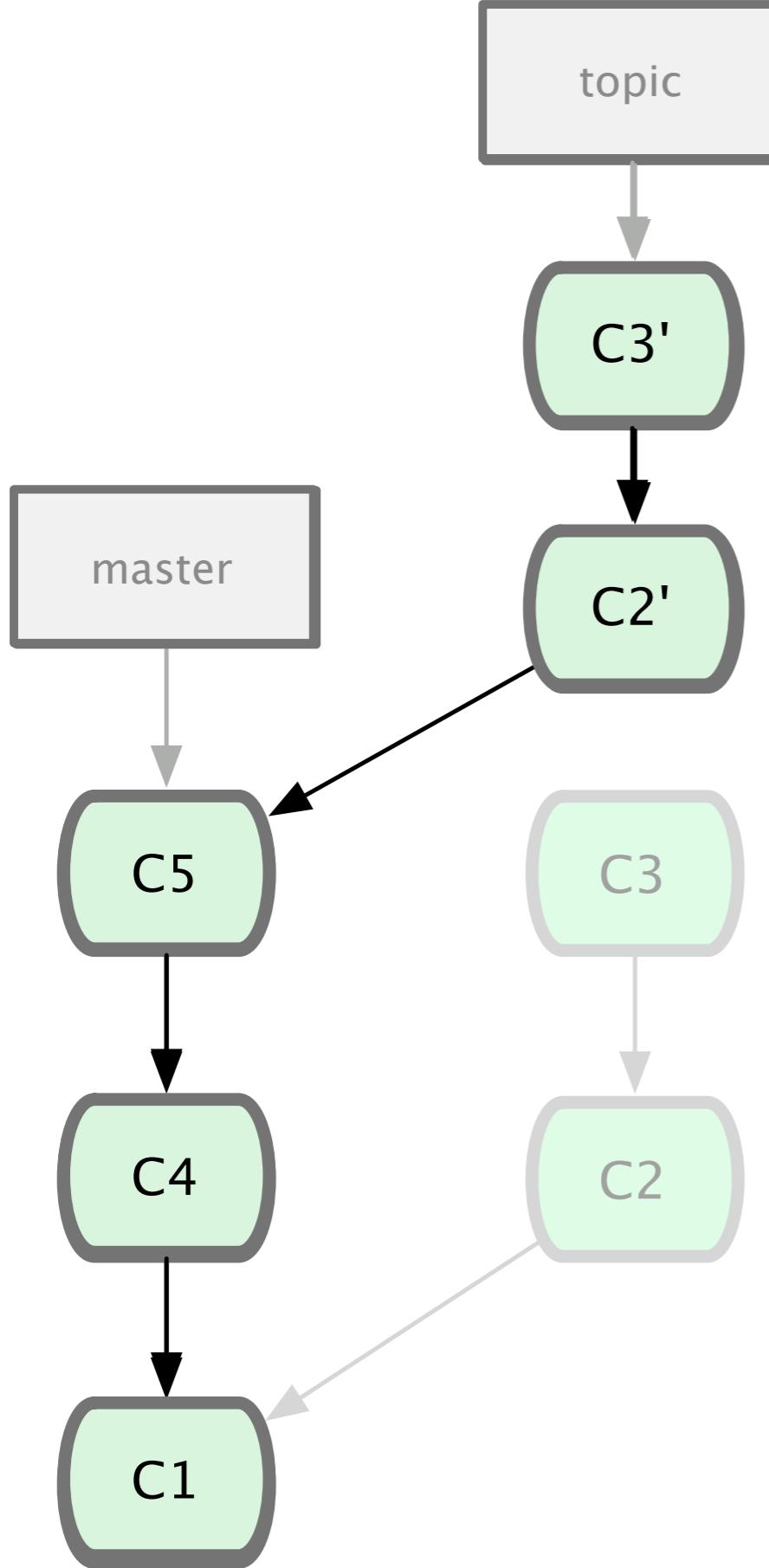




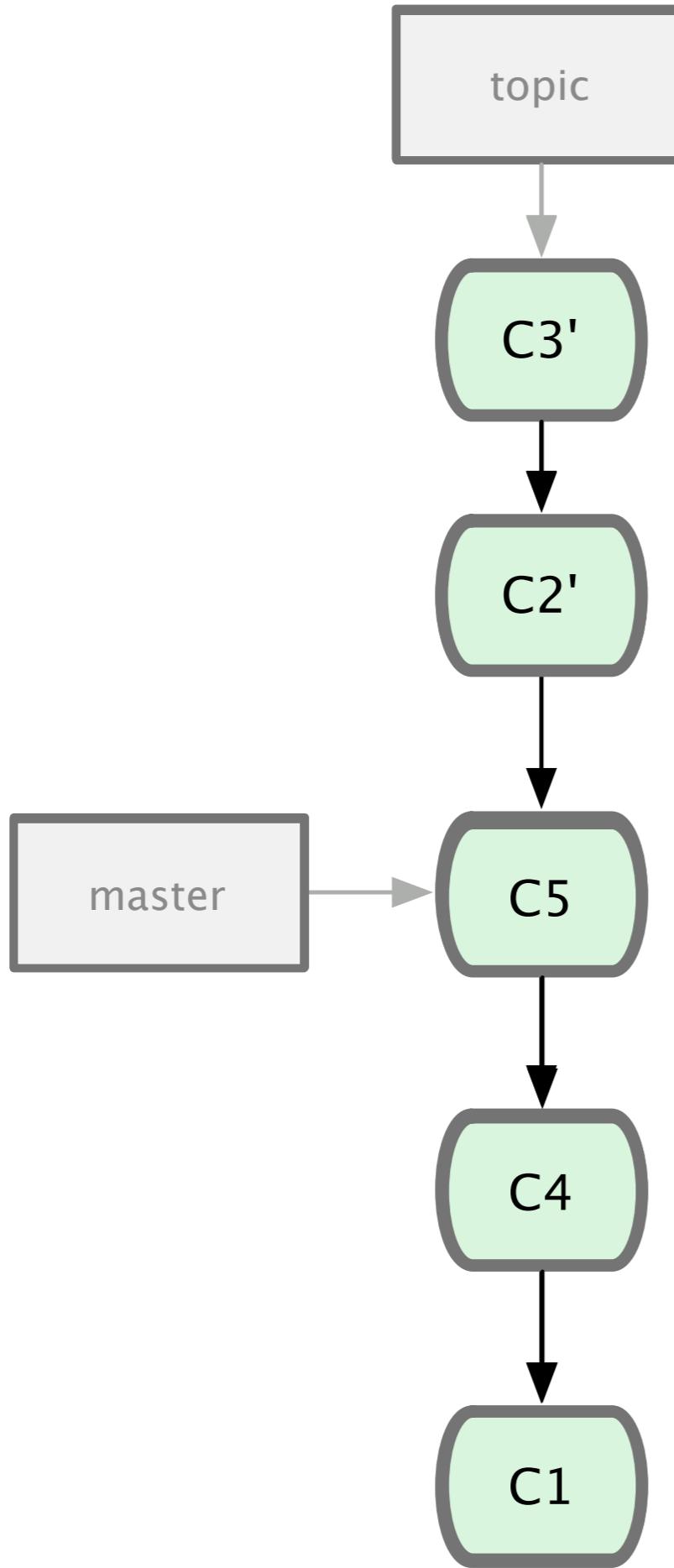
git rebase master



git rebase master



git rebase master

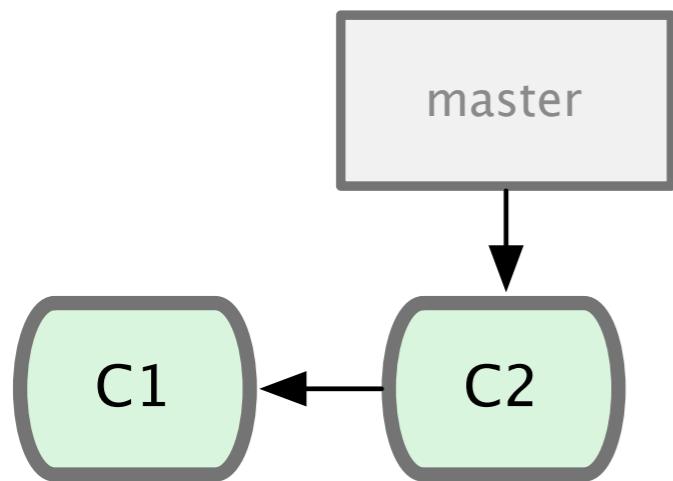


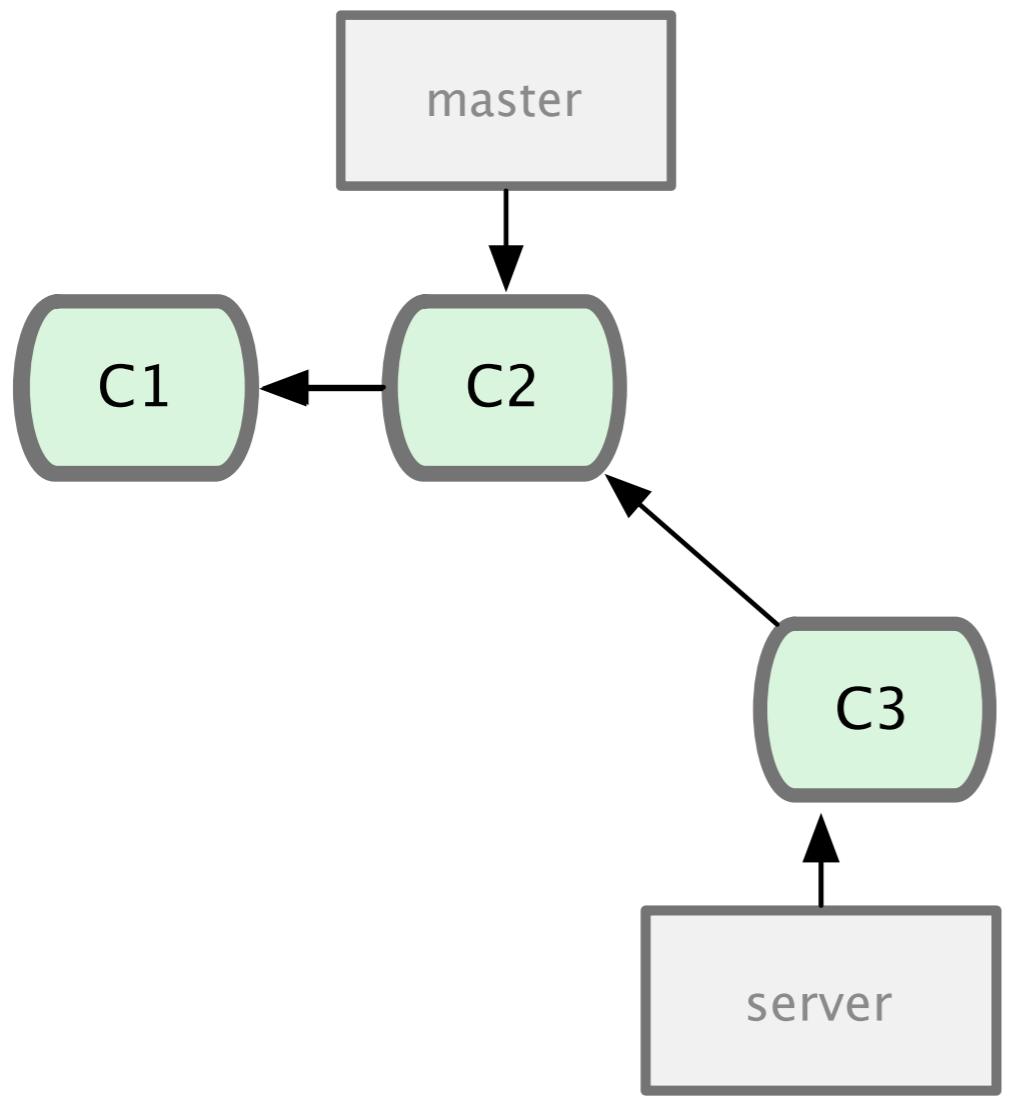
git rebase master

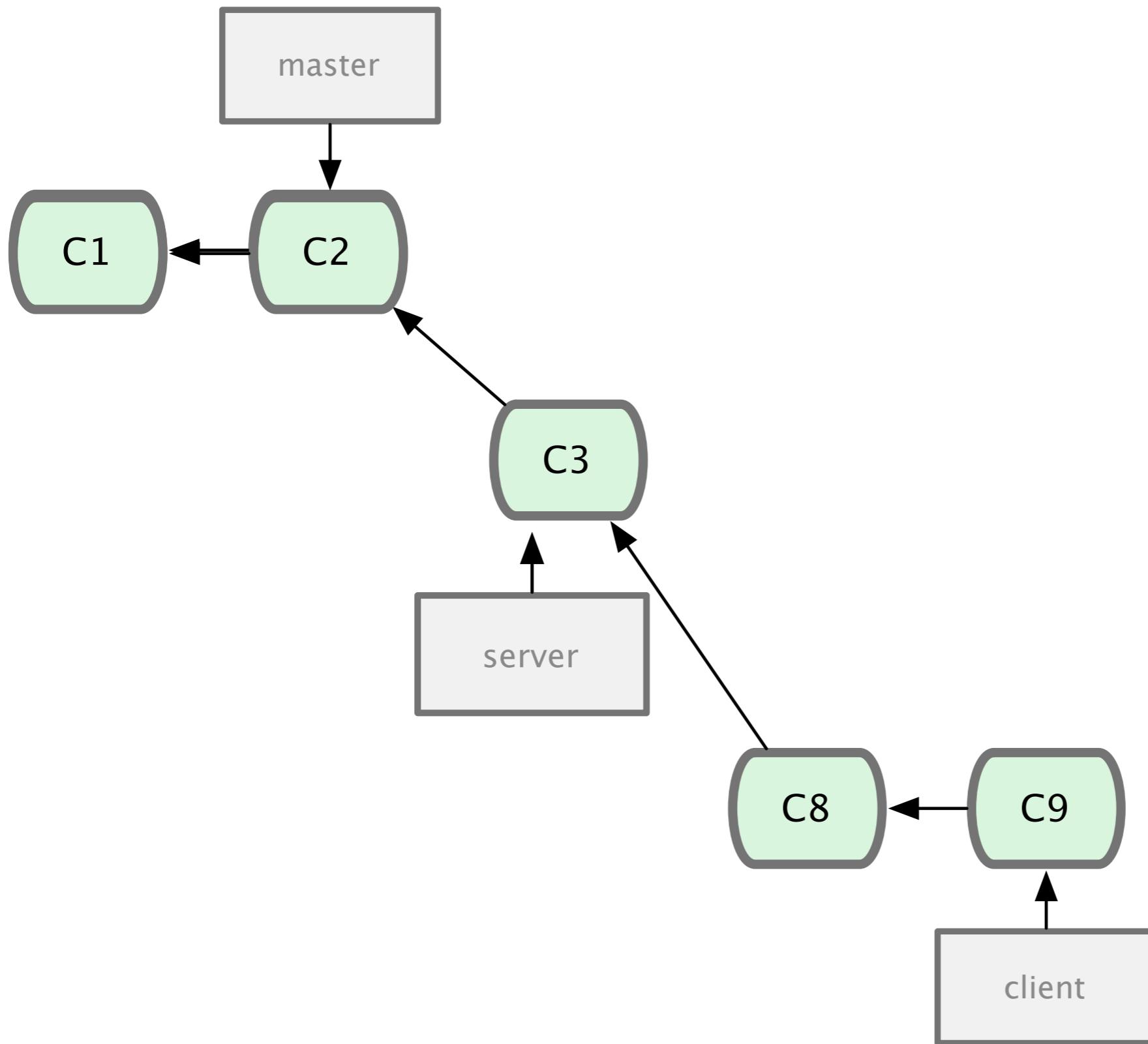
Fun with Rebasing

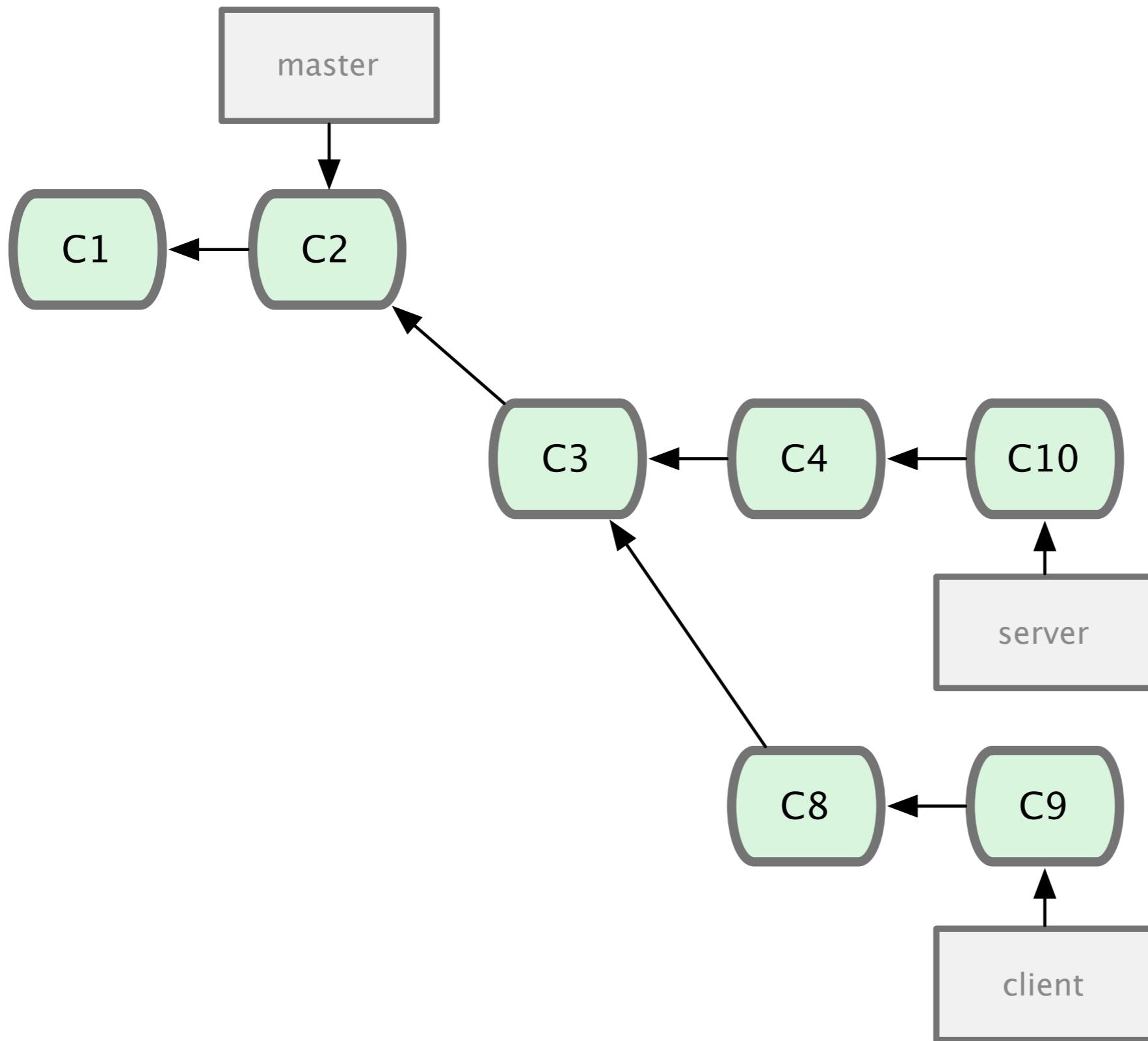
rebase --onto

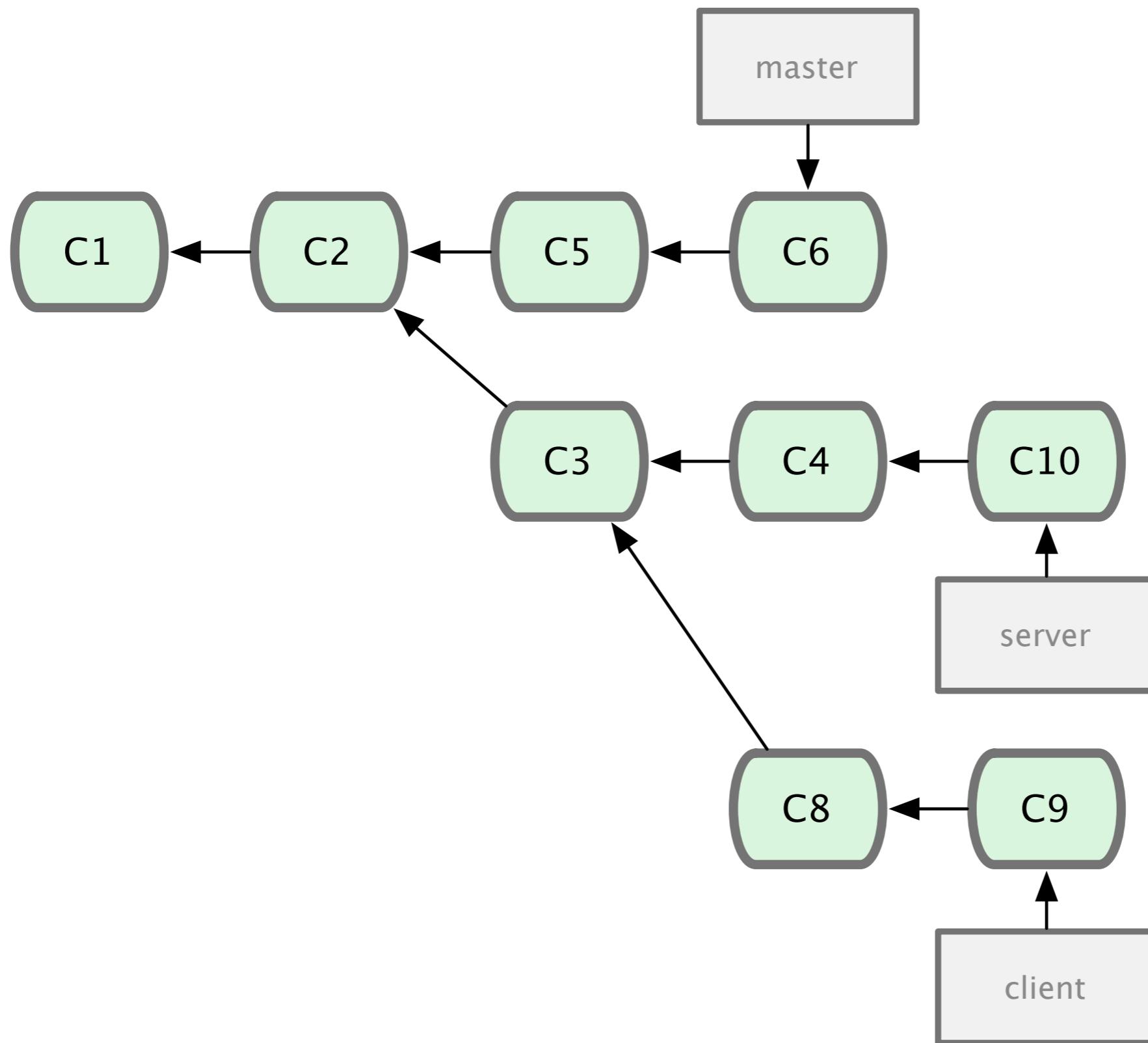
Transplanting Topic Branches



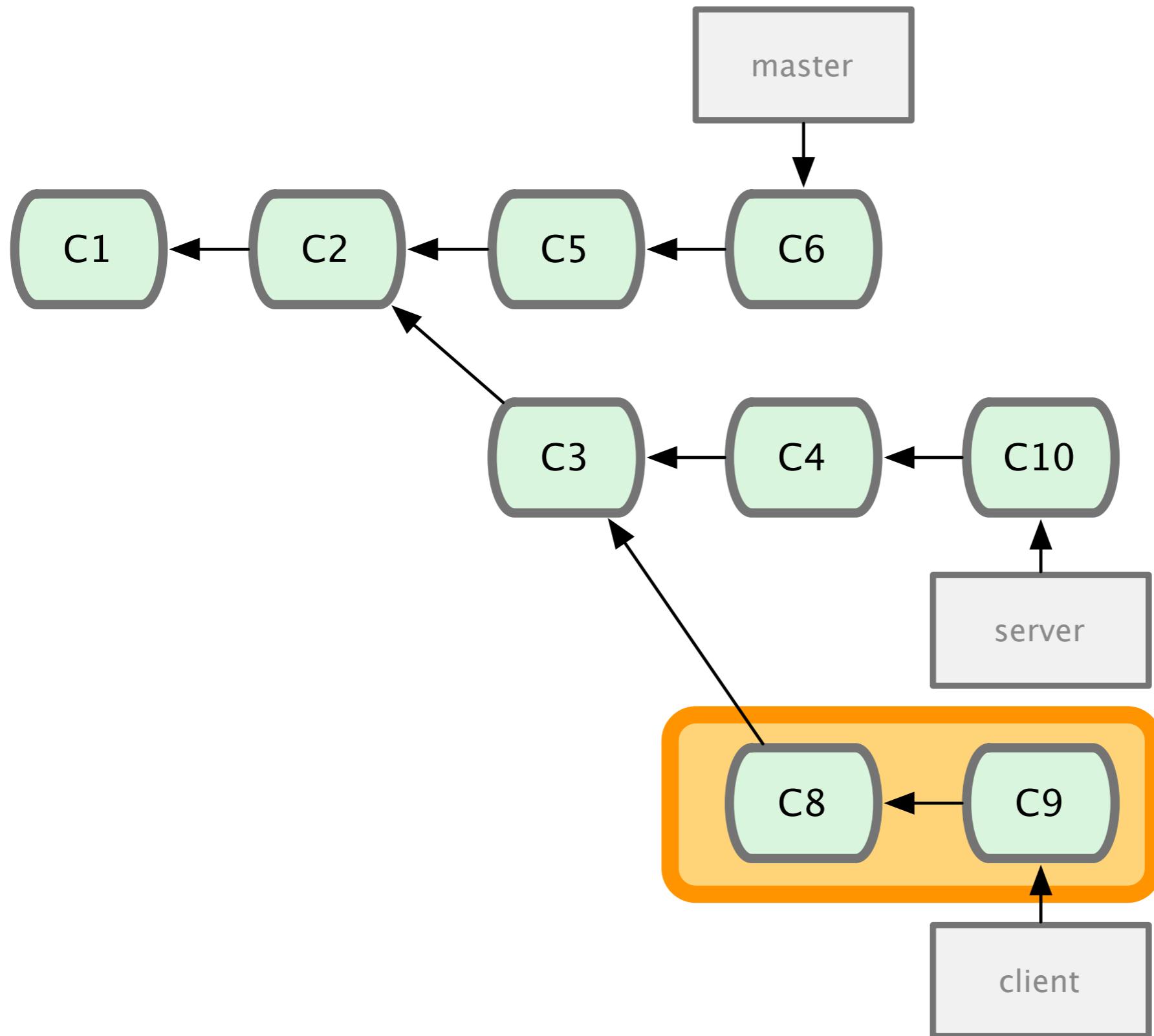


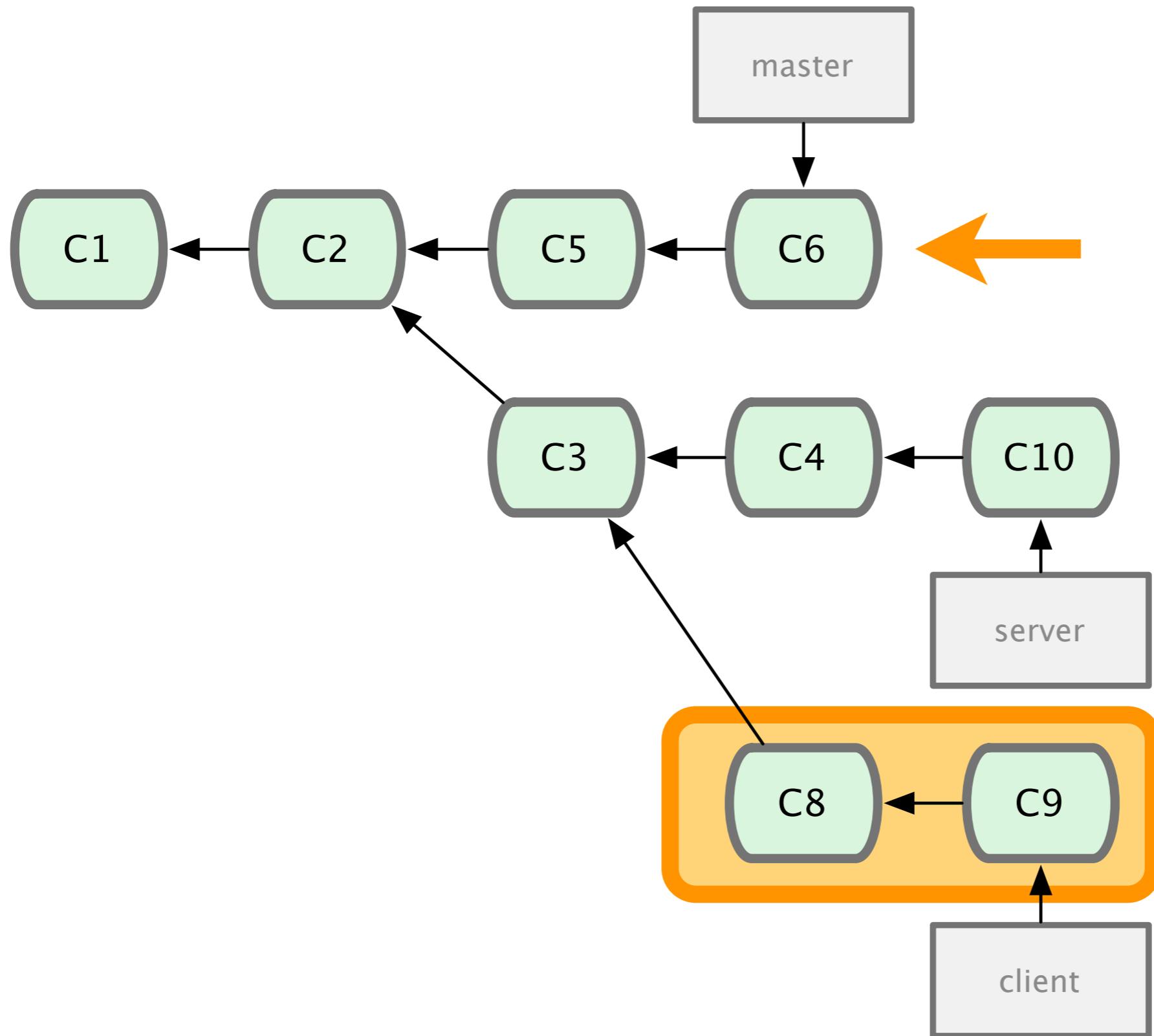


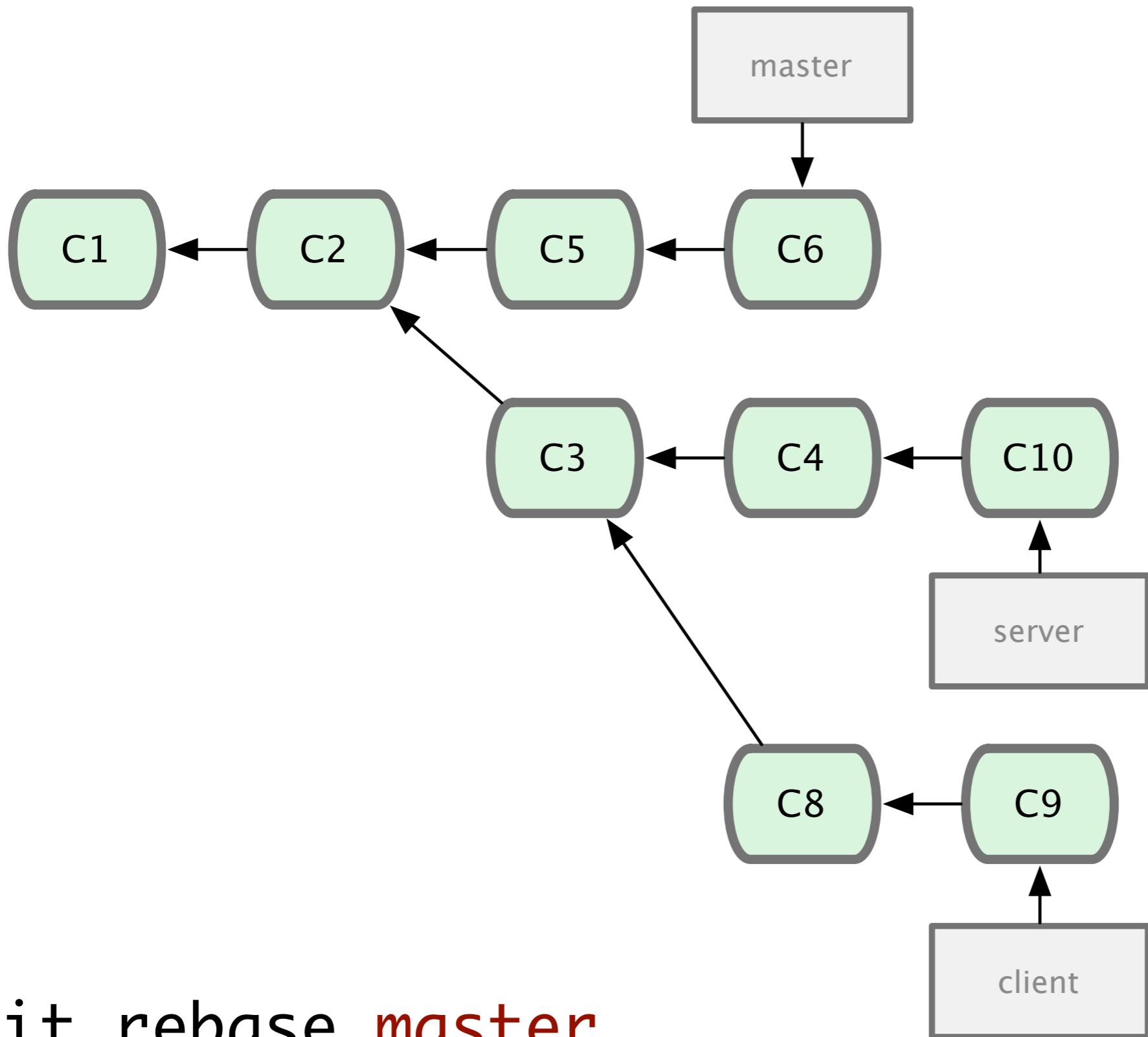




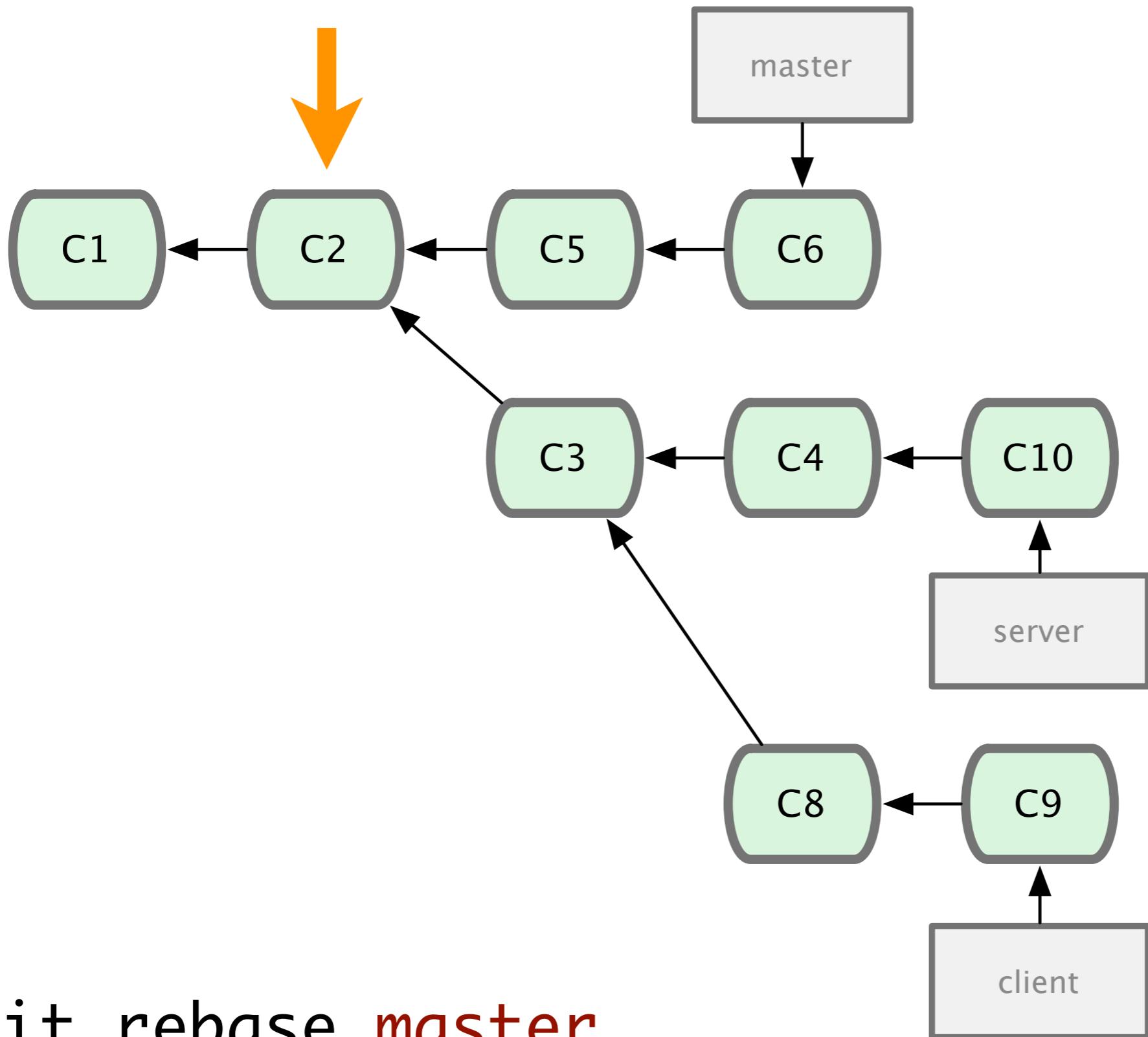
move your ‘client’
branch work to your
‘master’ branch



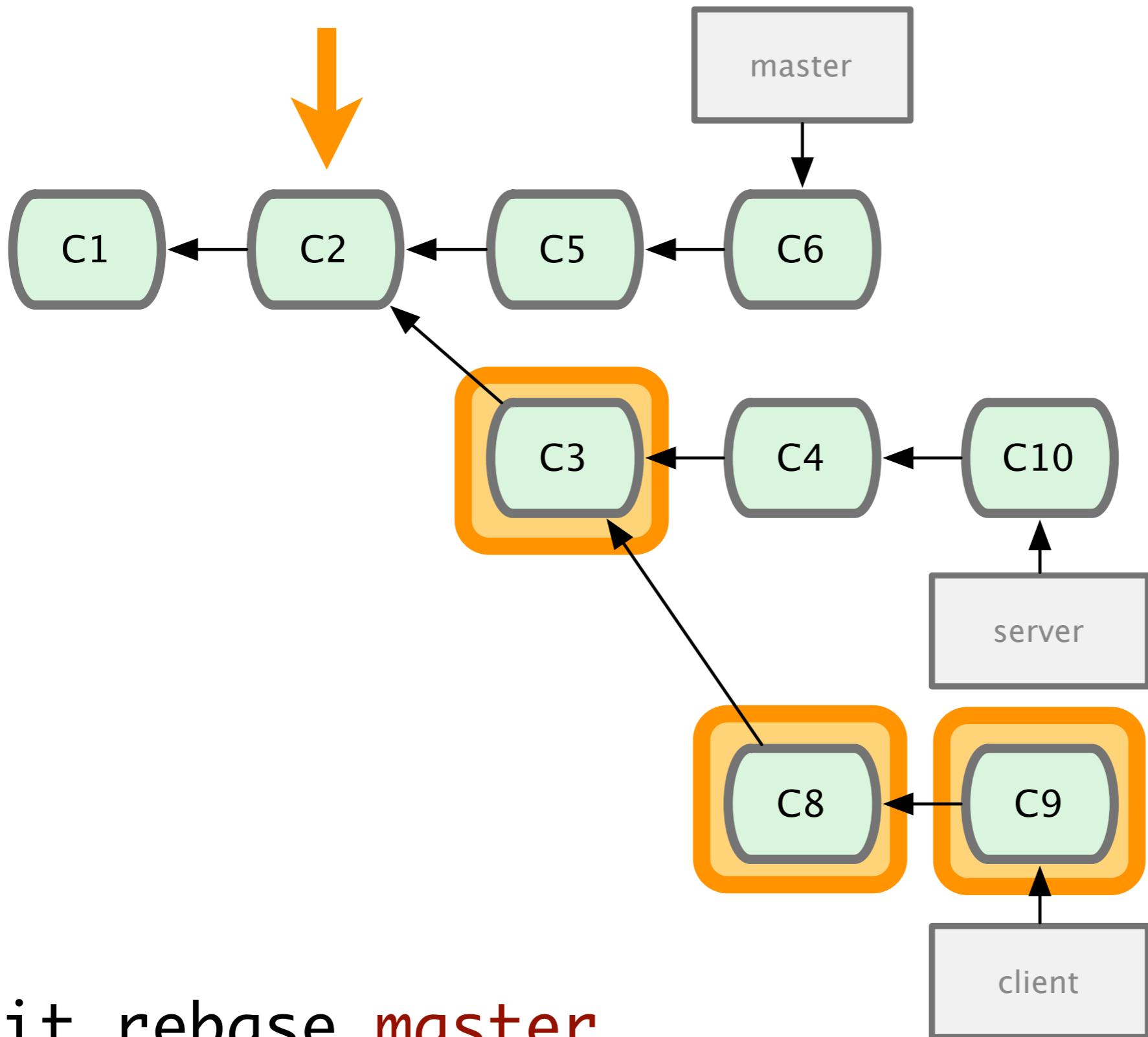




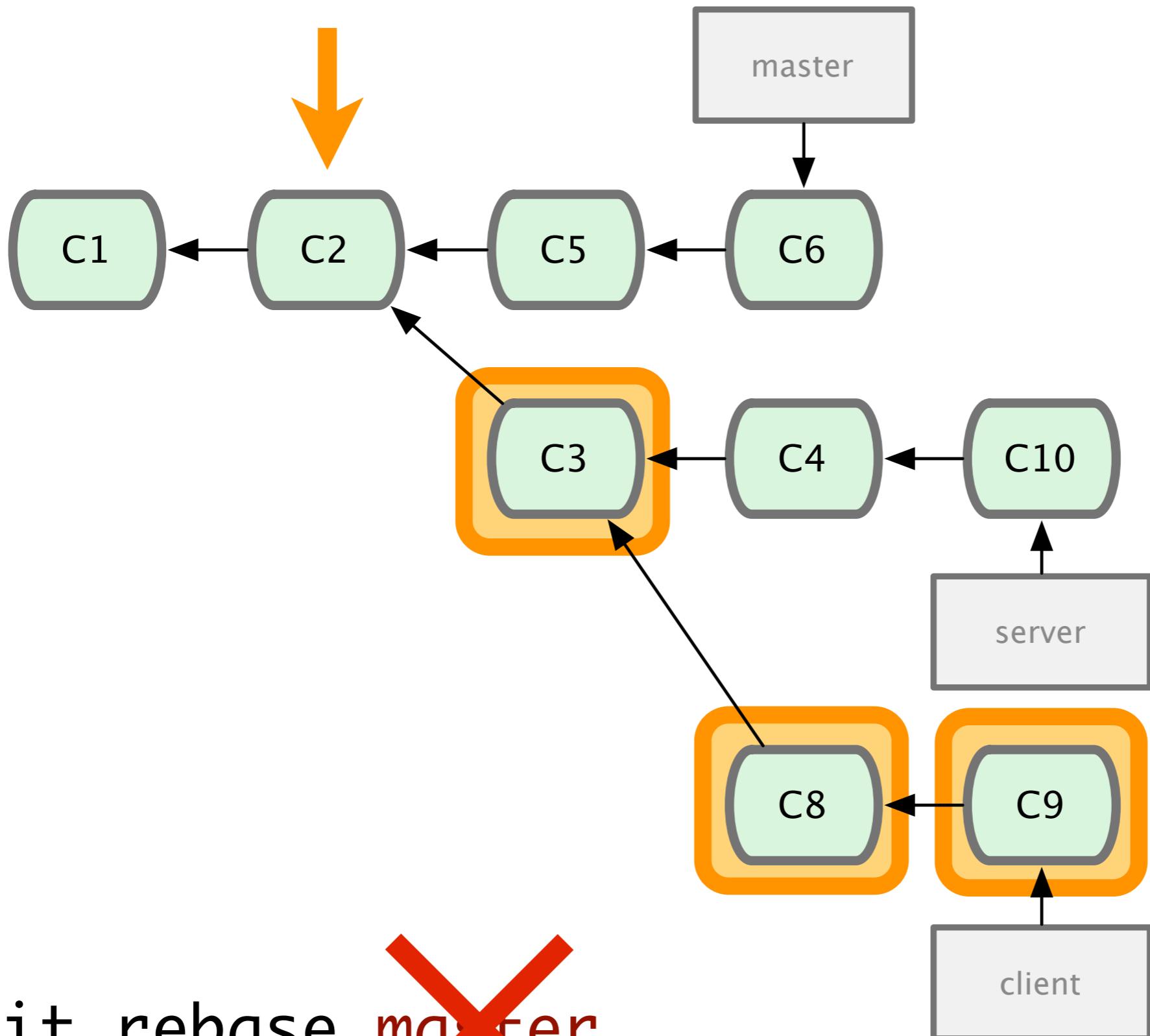
git rebase master



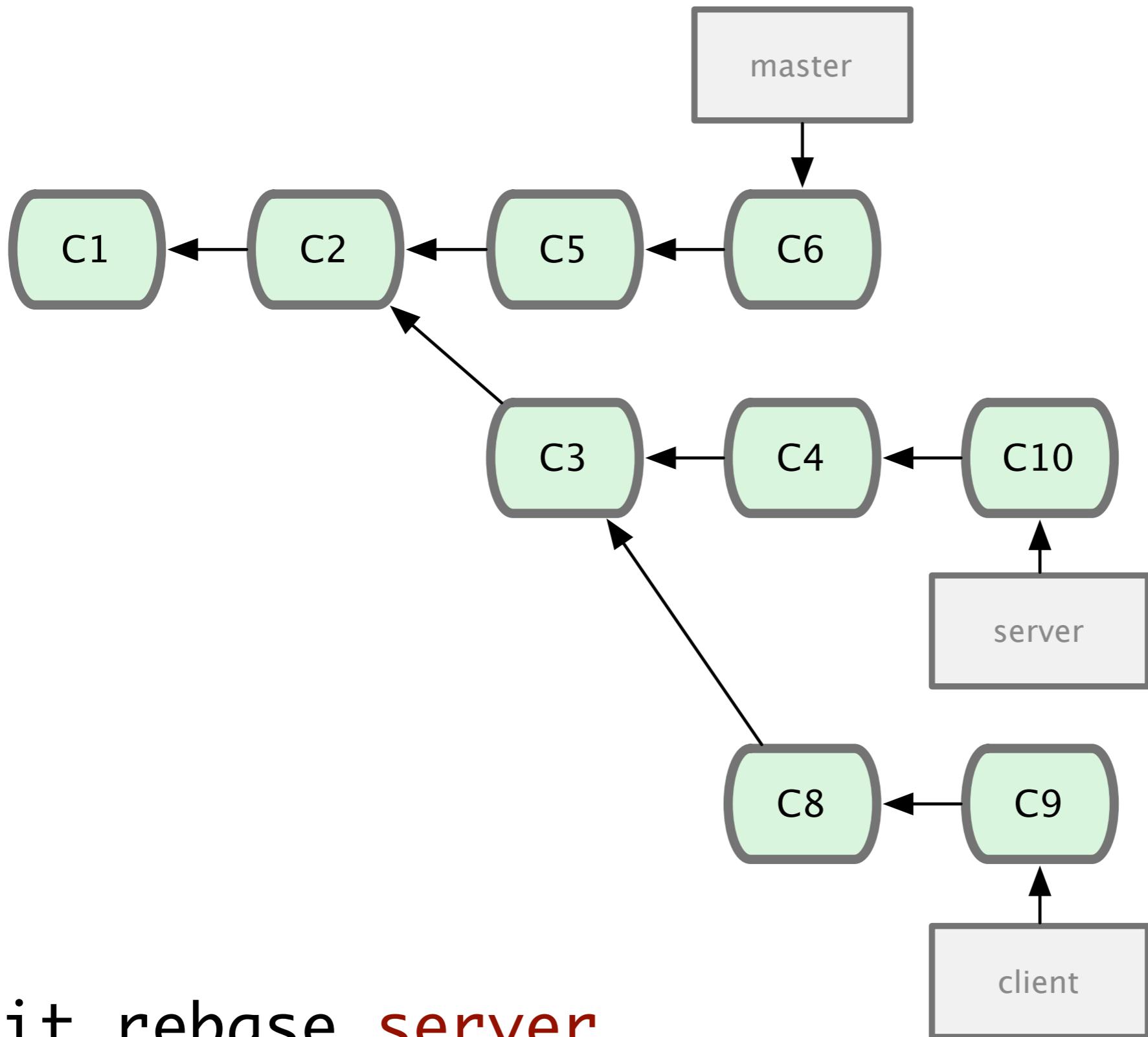
git rebase master



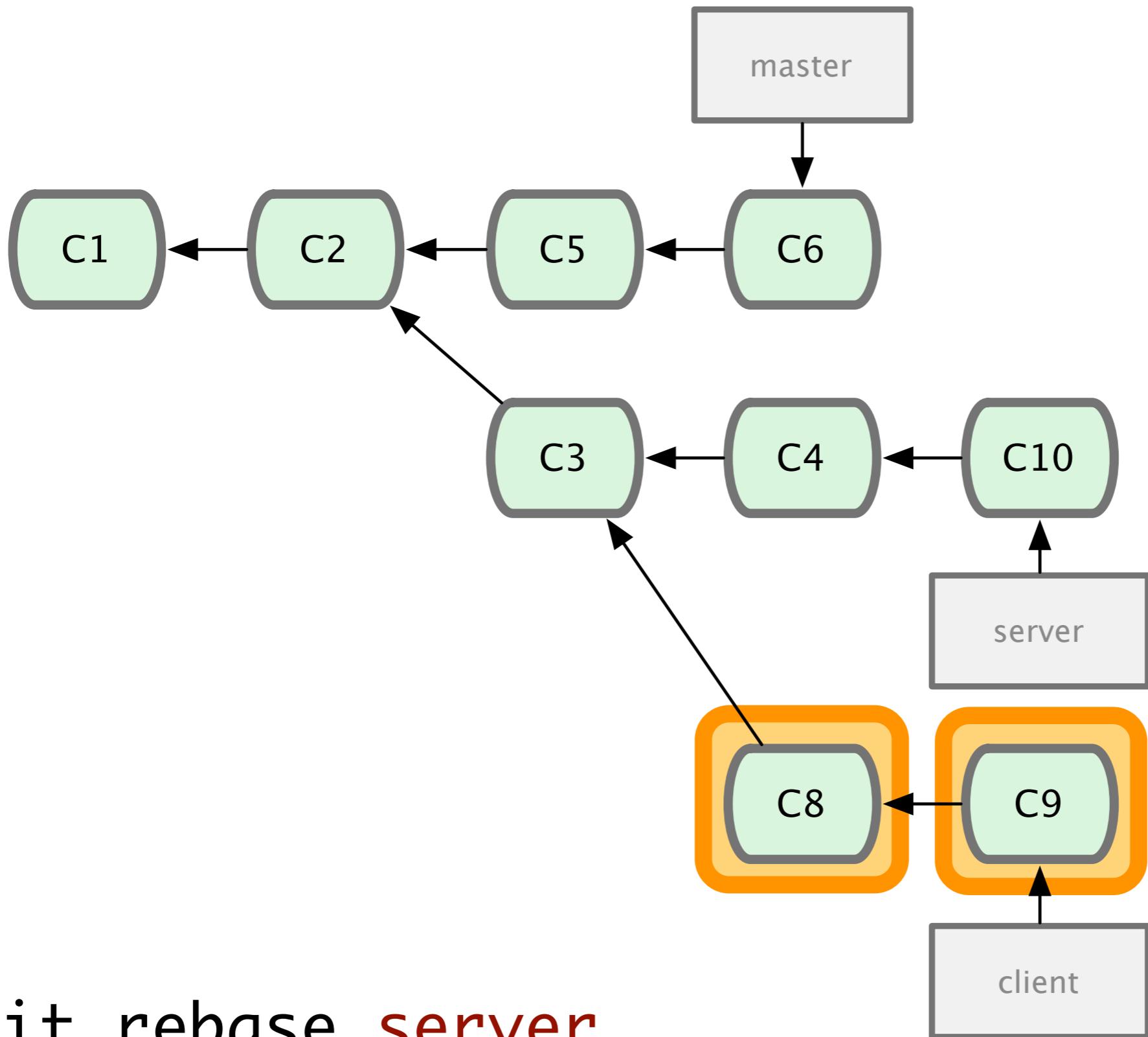
git rebase master



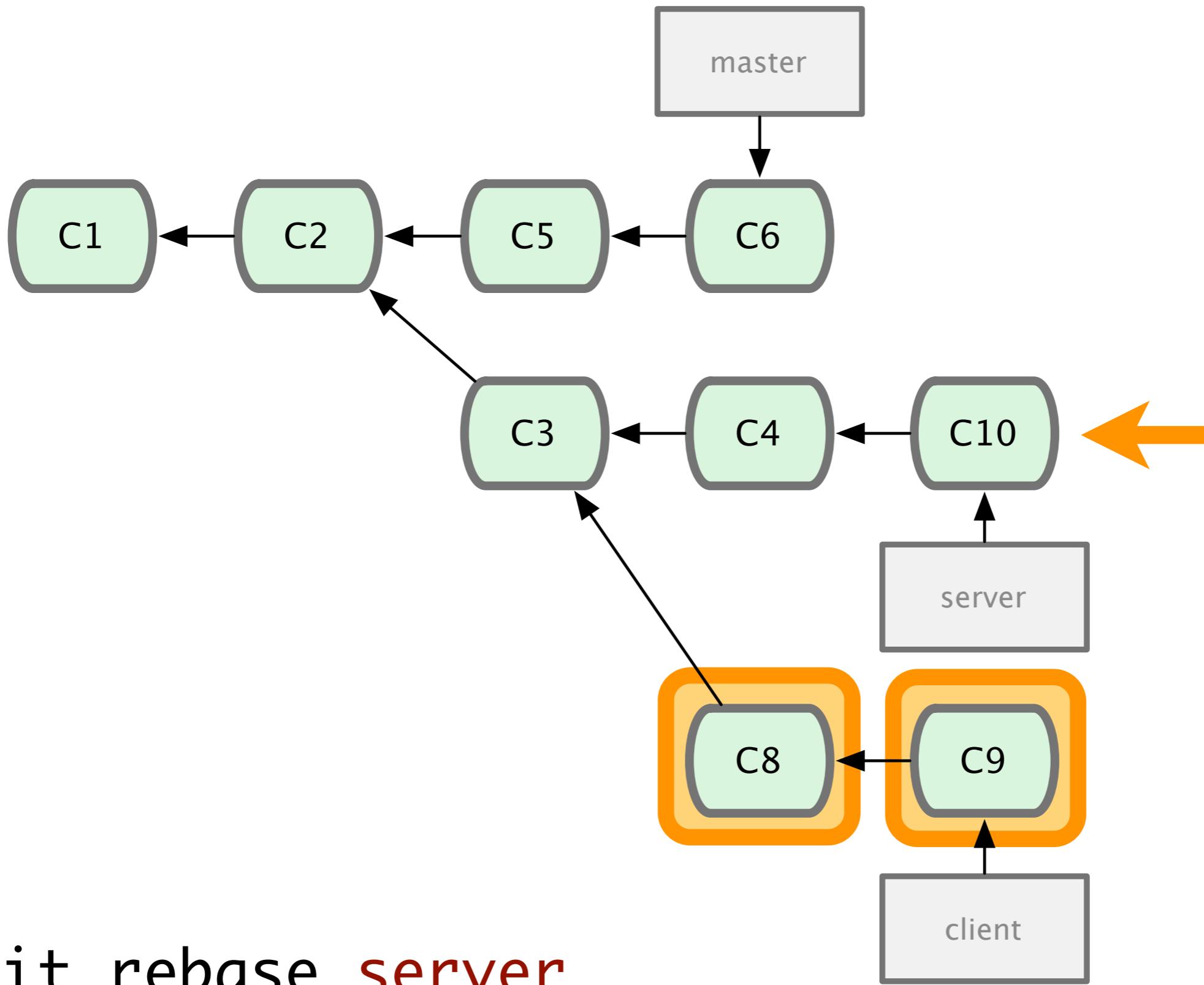
git rebase master



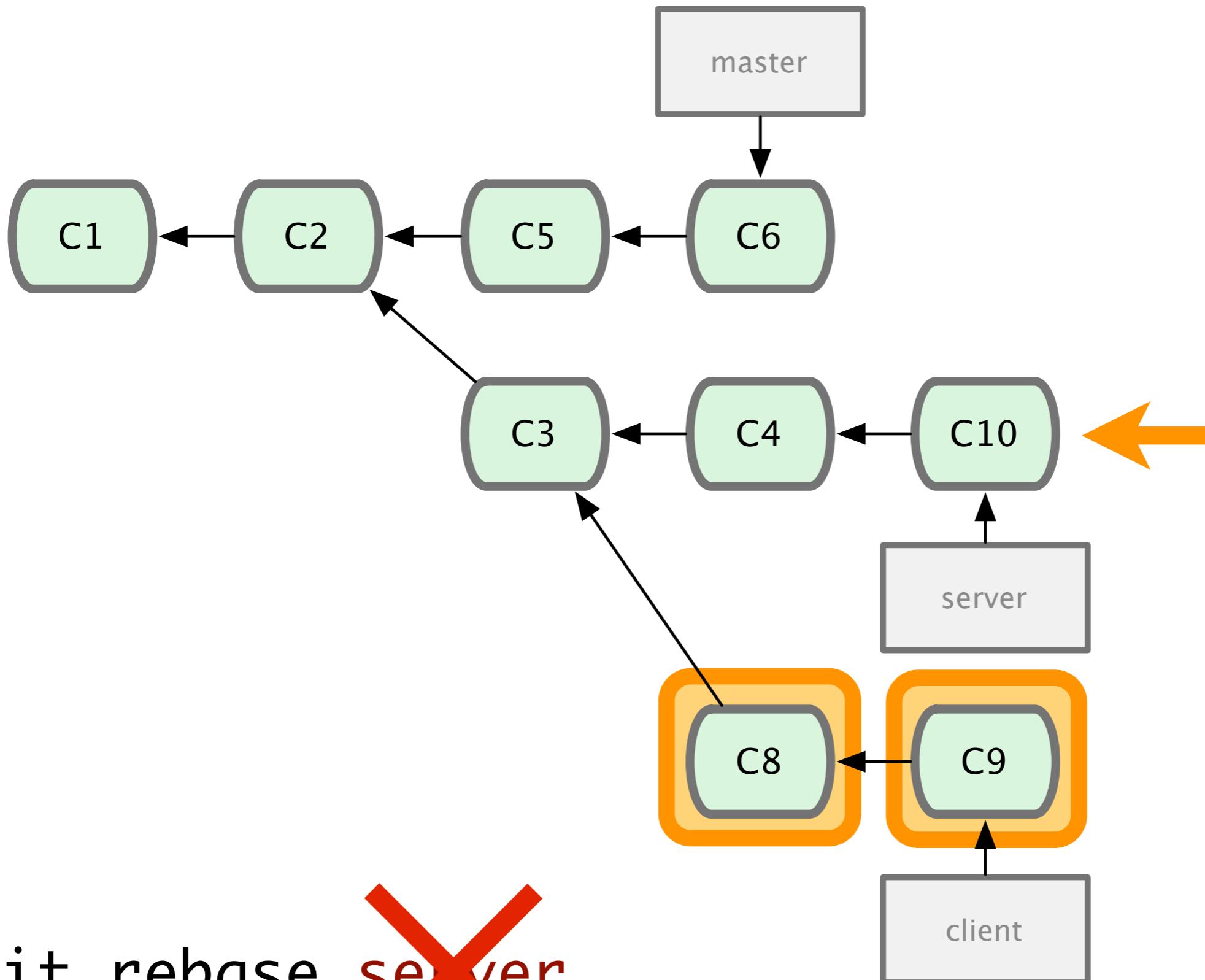
git rebase server



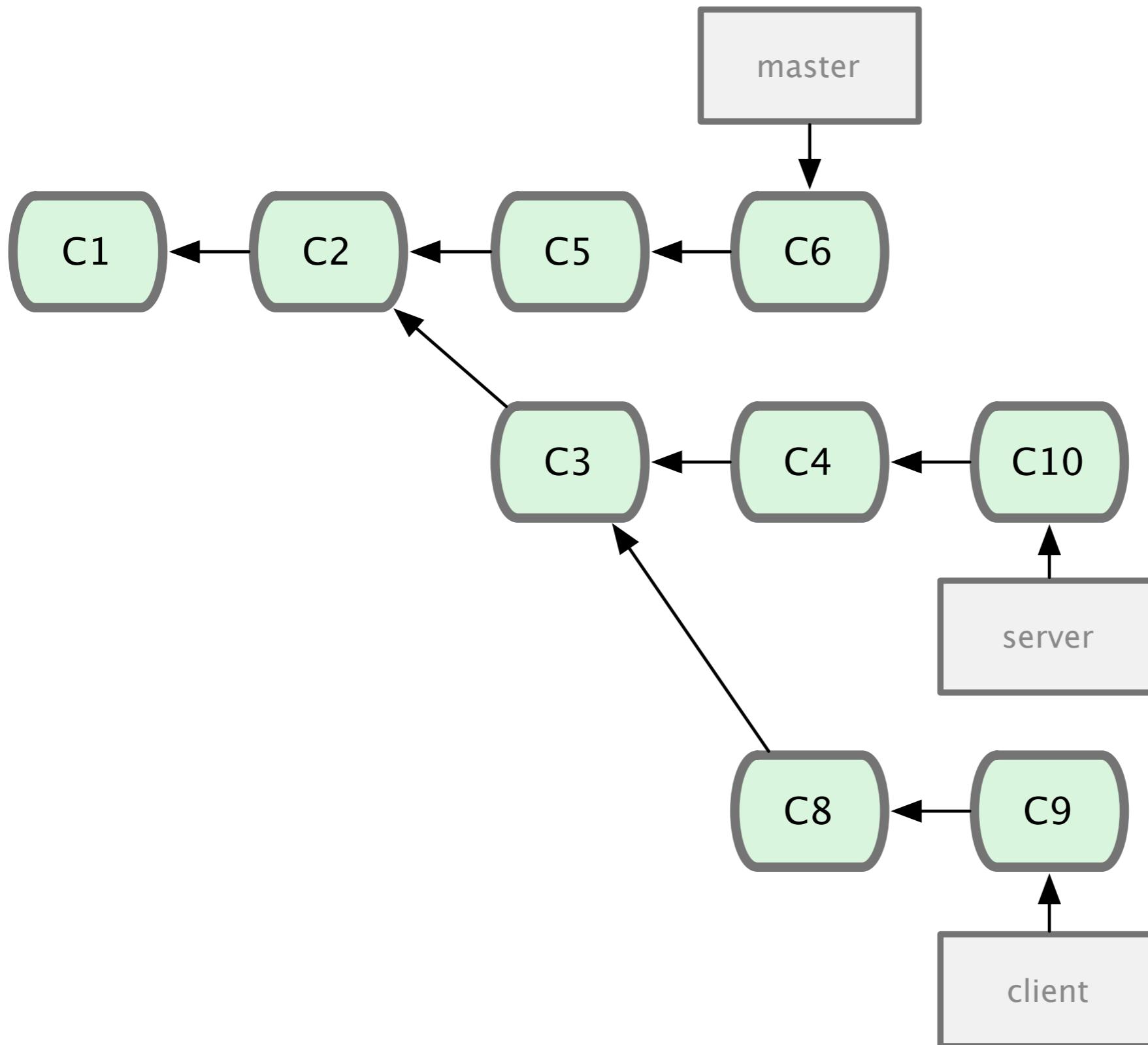
git rebase server



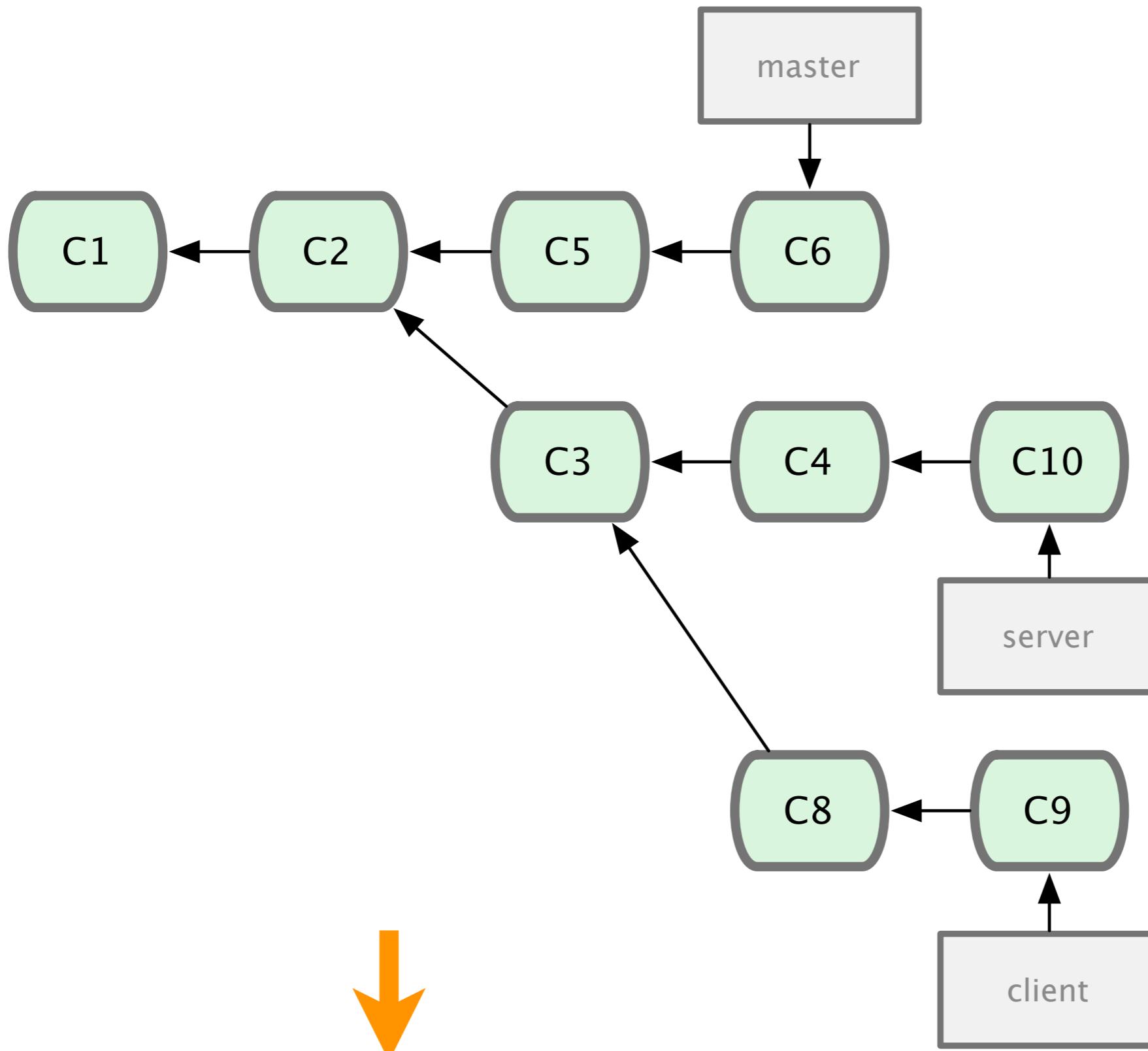
git rebase server



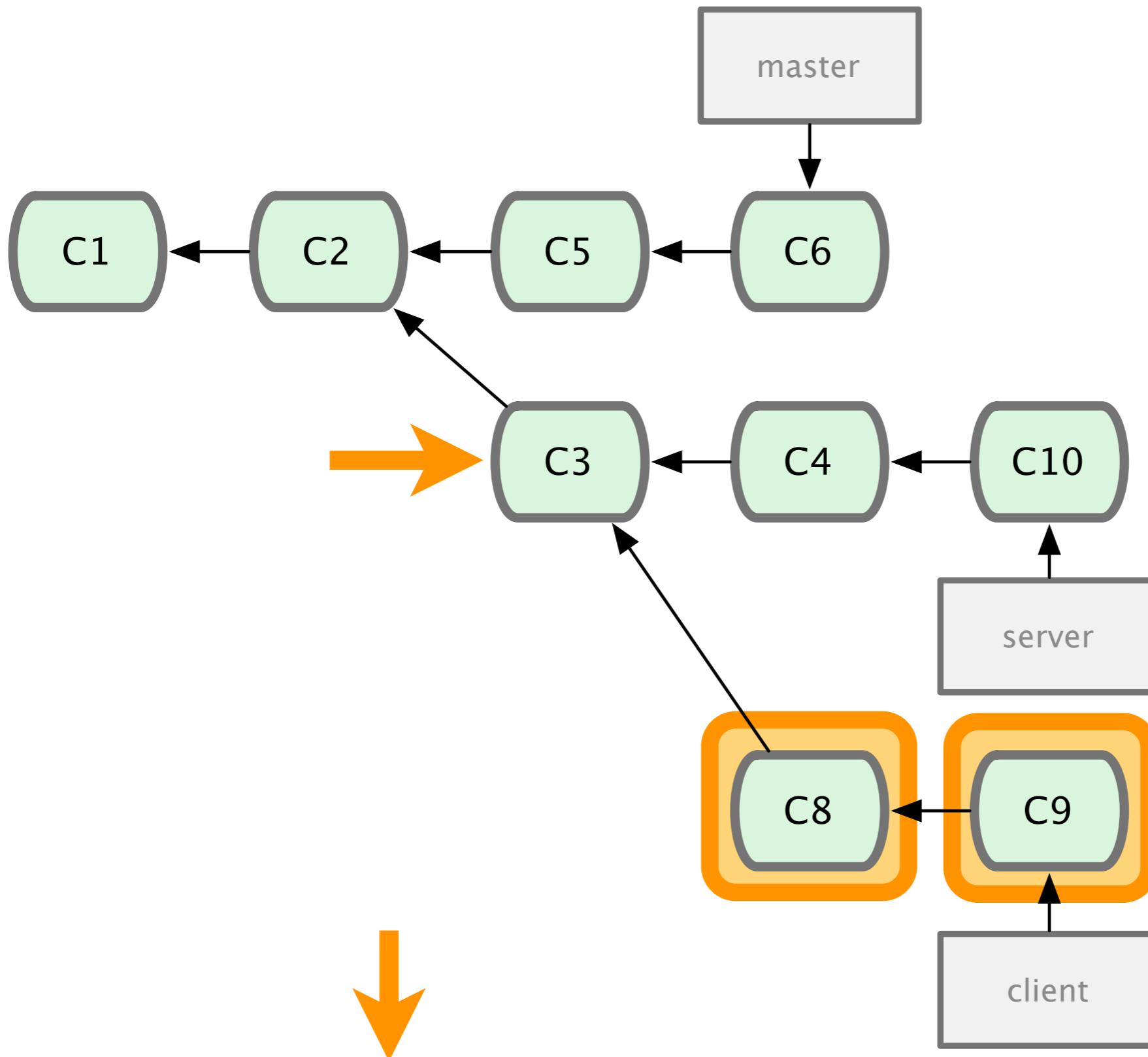
git rebase ~~server~~



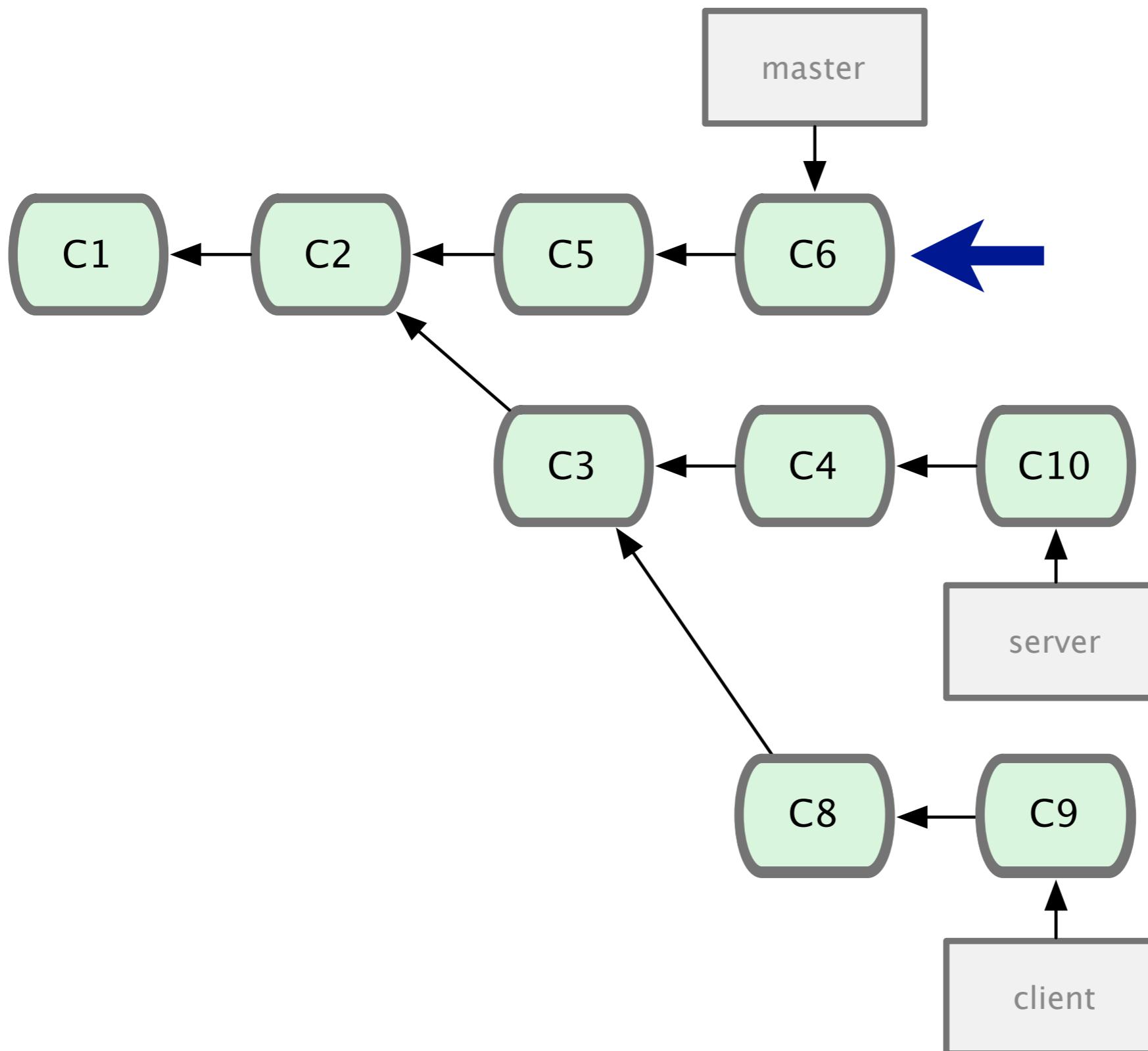
git rebase server --onto master



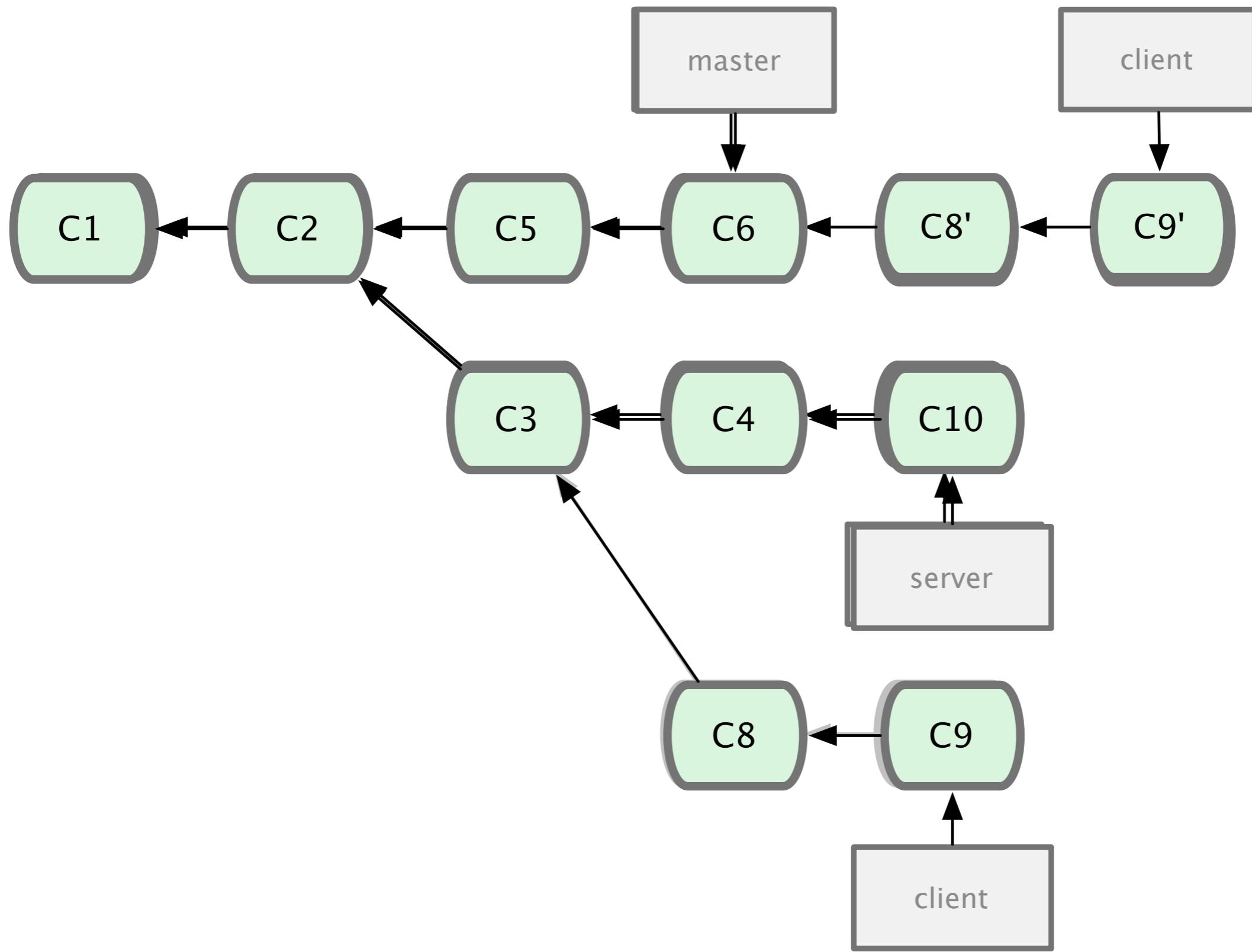
git rebase server --onto master

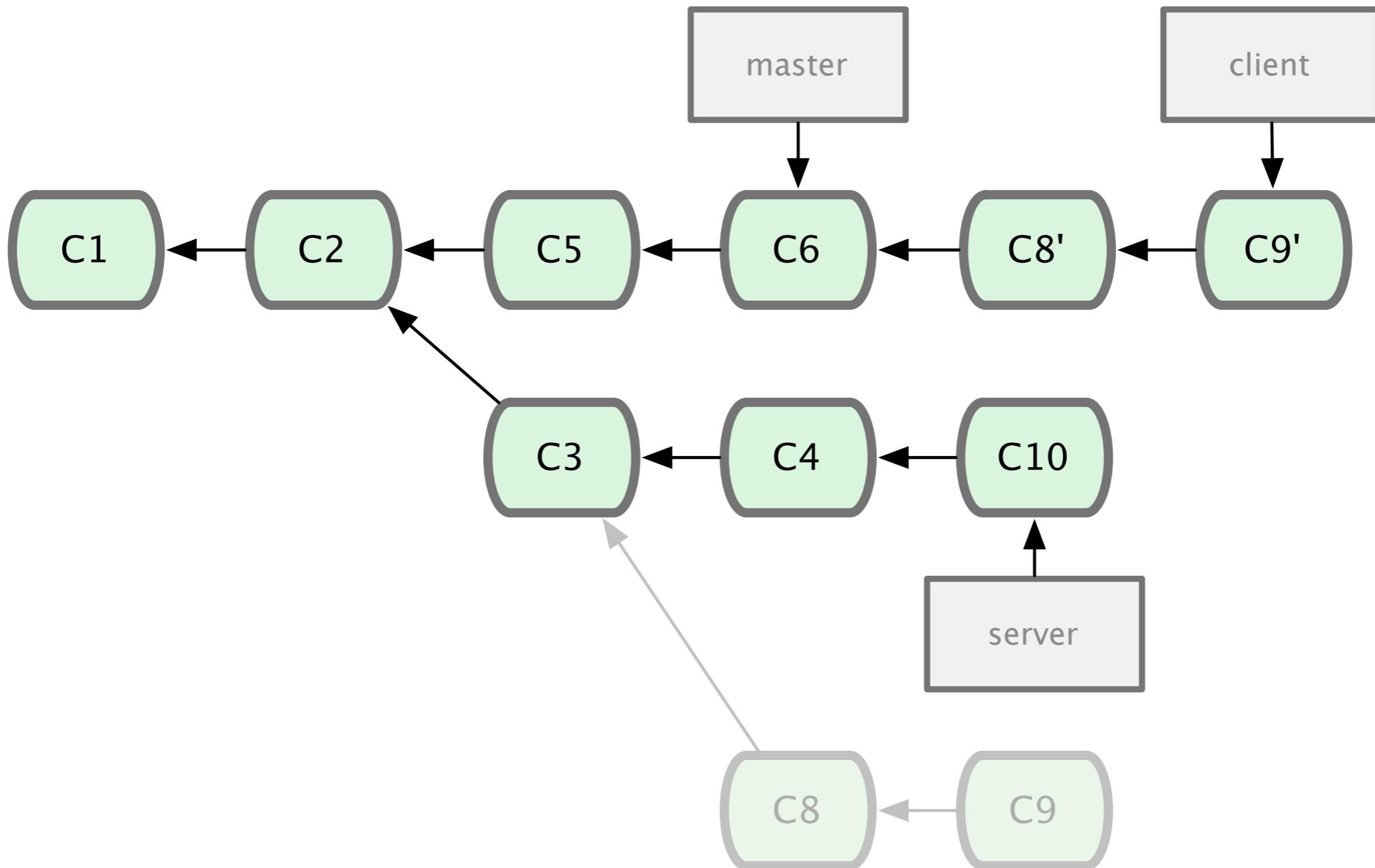


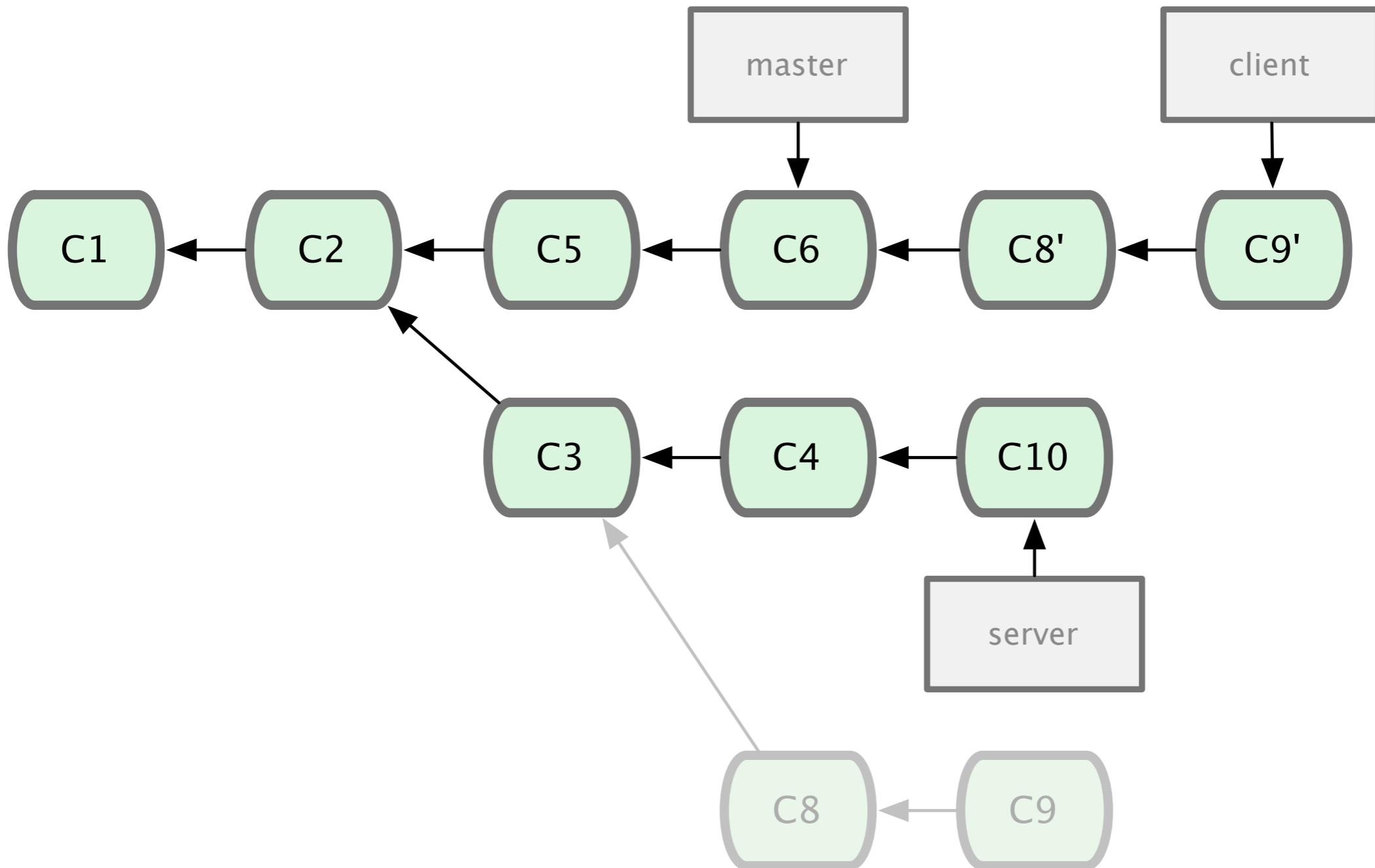
git rebase server --onto master



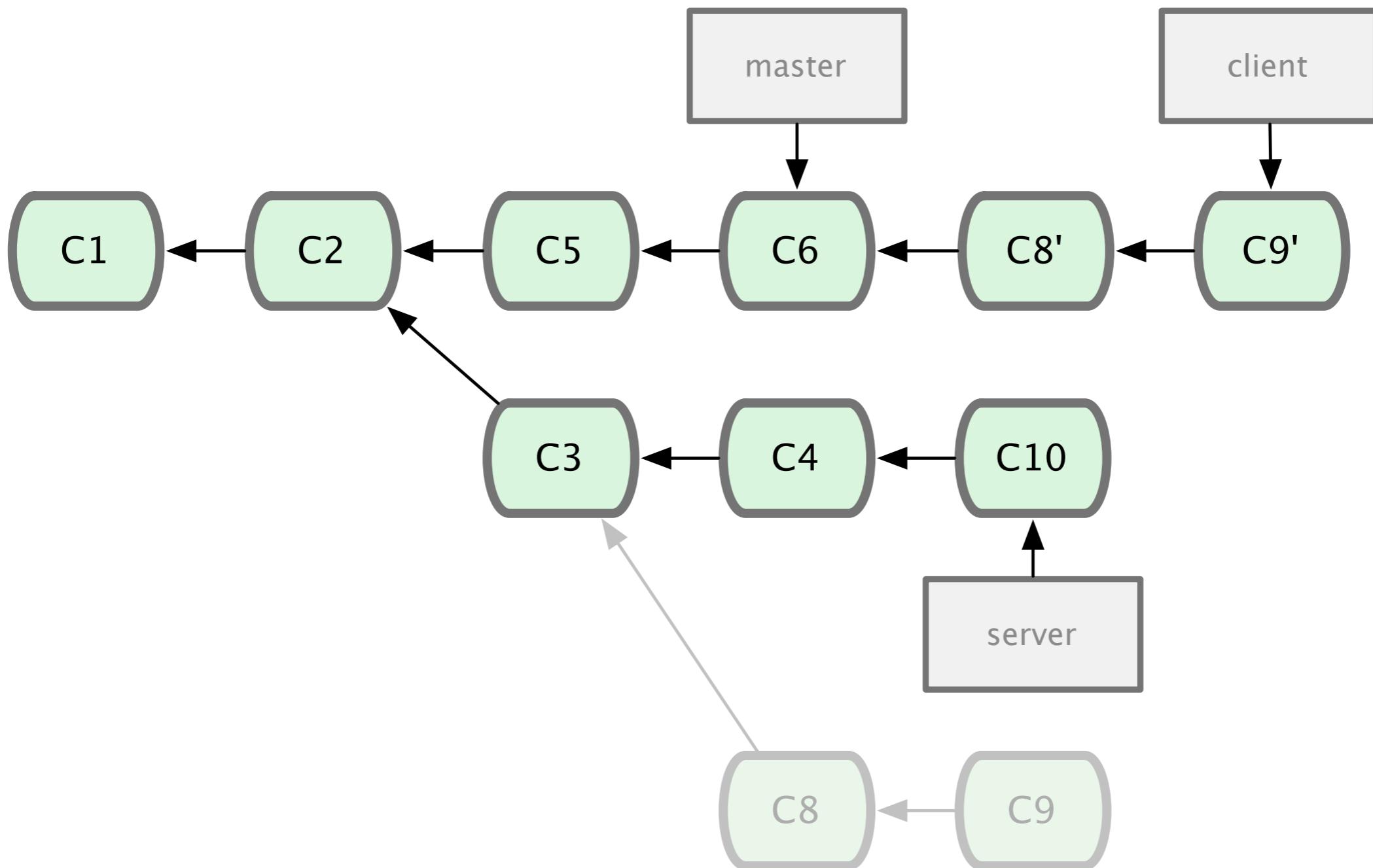
git rebase server --onto master ←



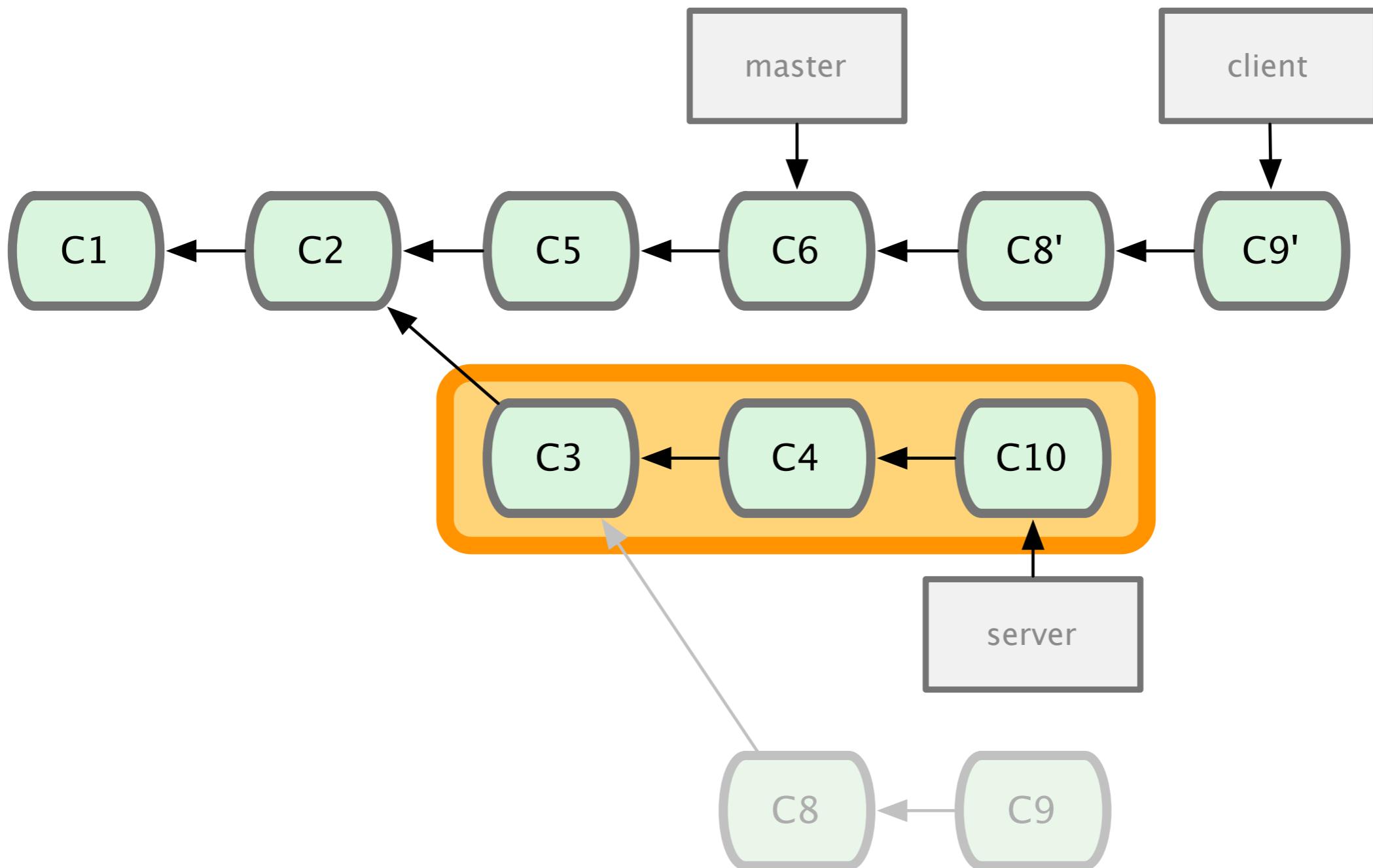




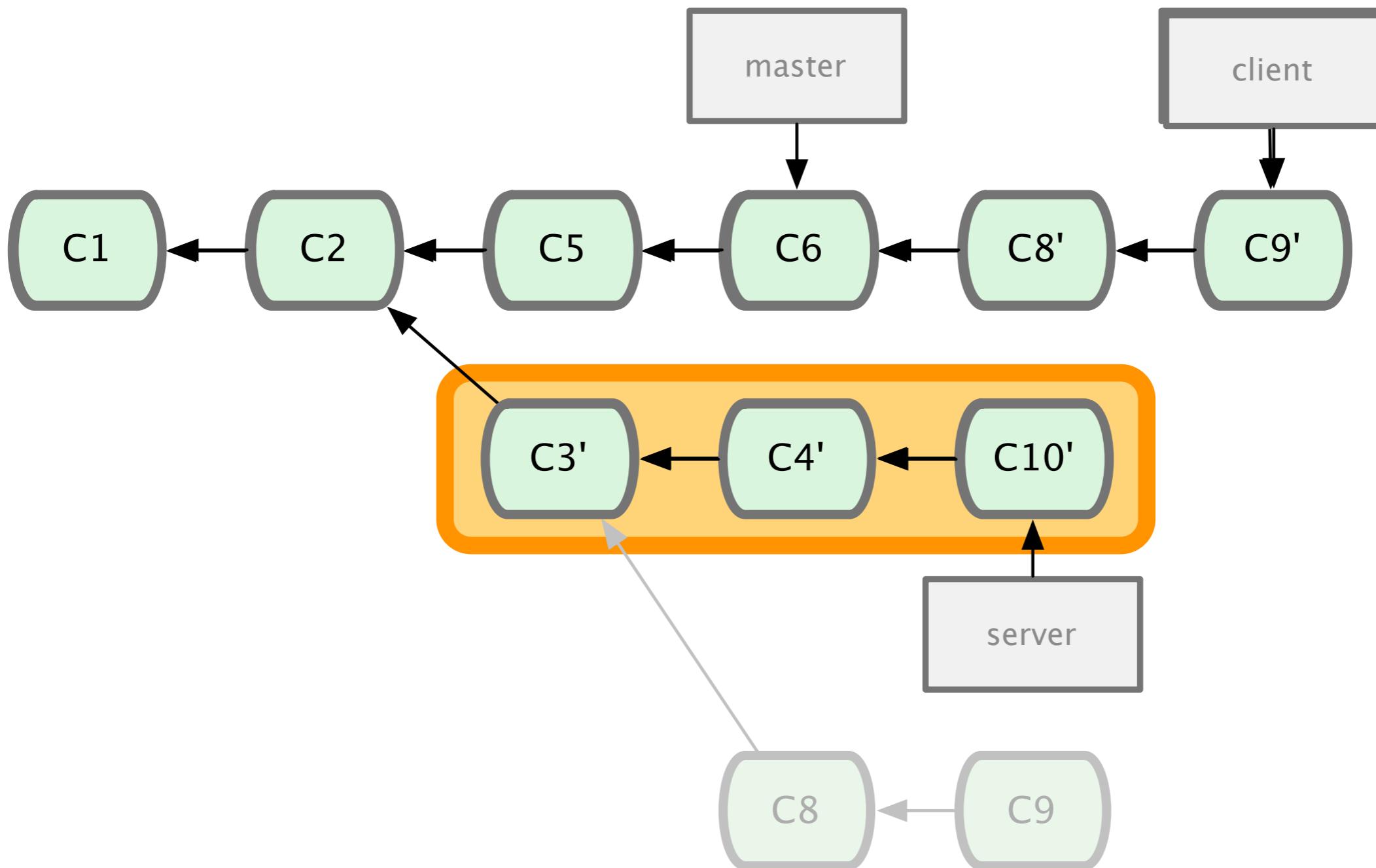
git checkout server



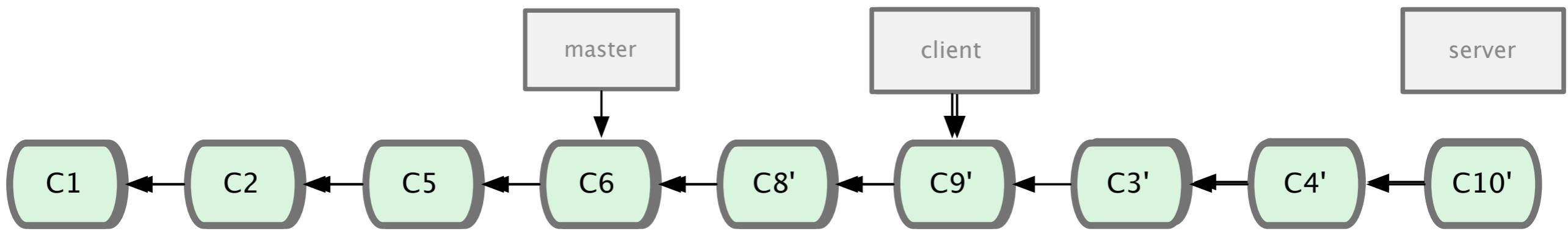
git checkout server
git rebase client



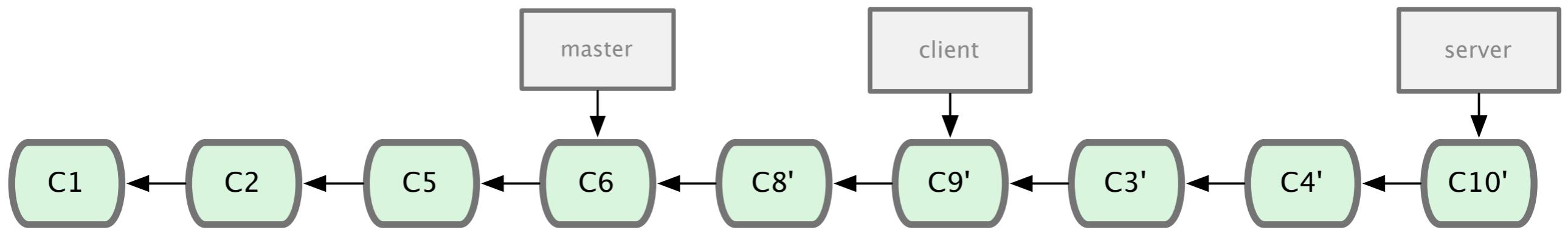
git checkout **server**
git rebase **client**



git checkout server
git rebase client

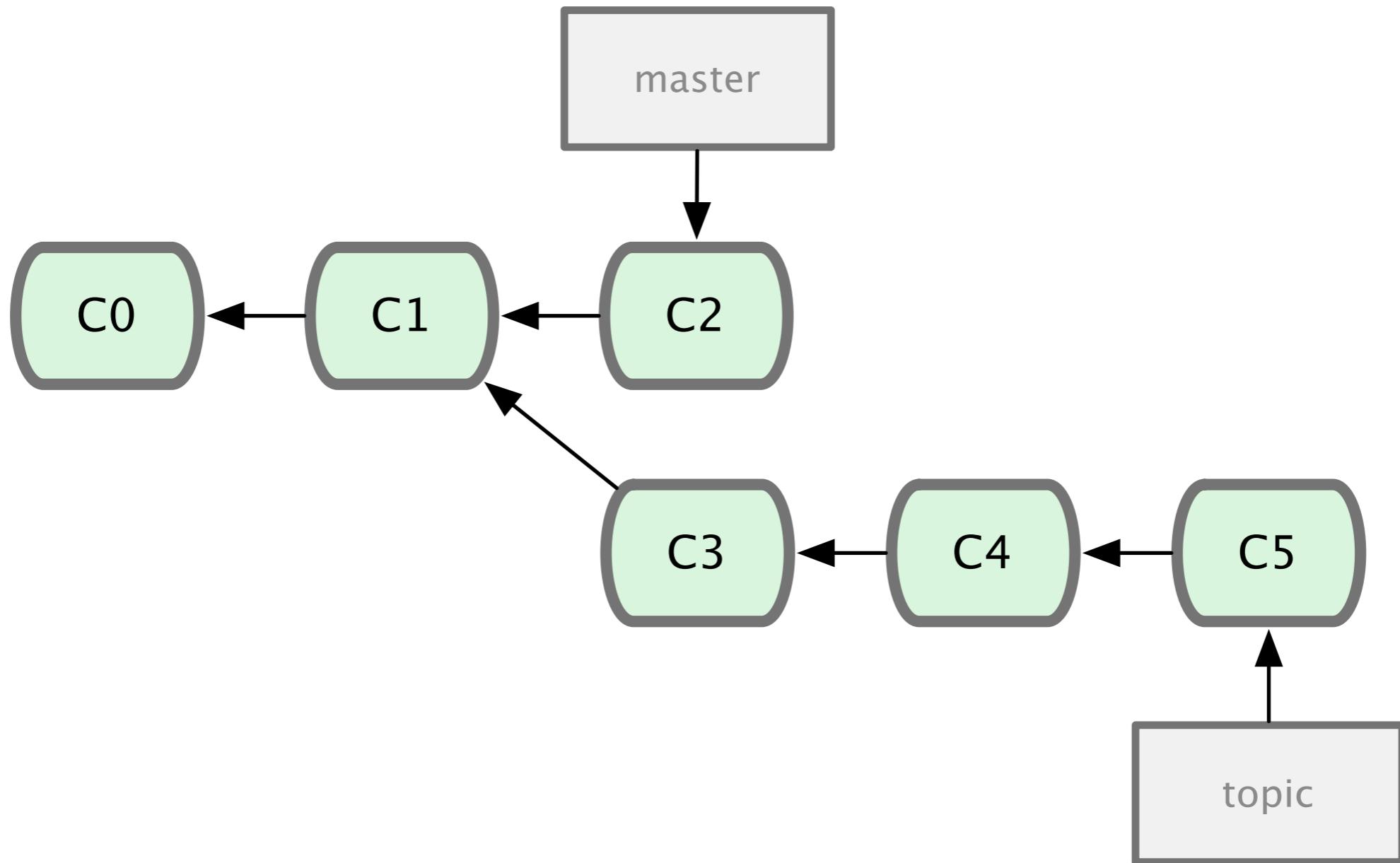


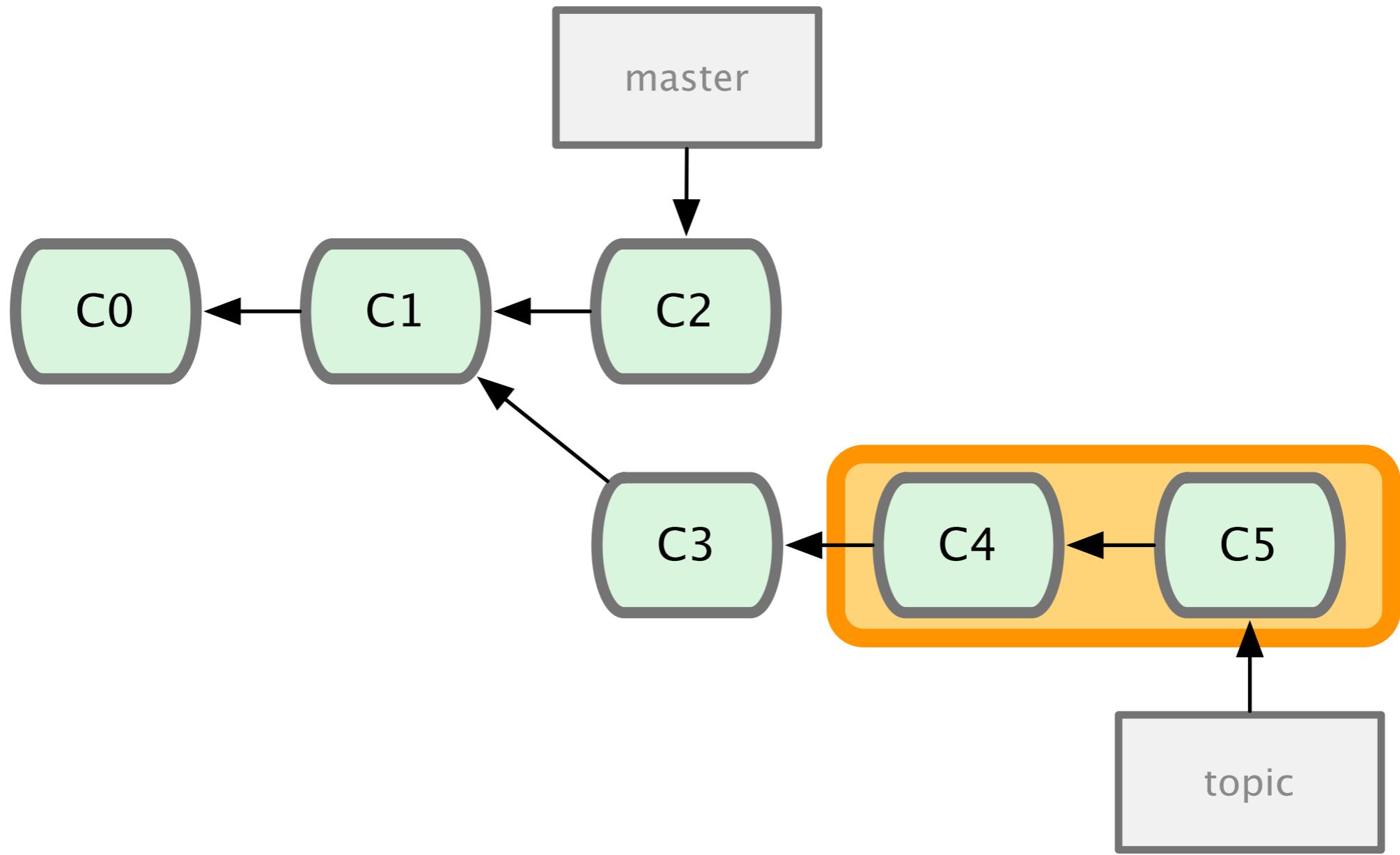
git checkout server
git rebase client

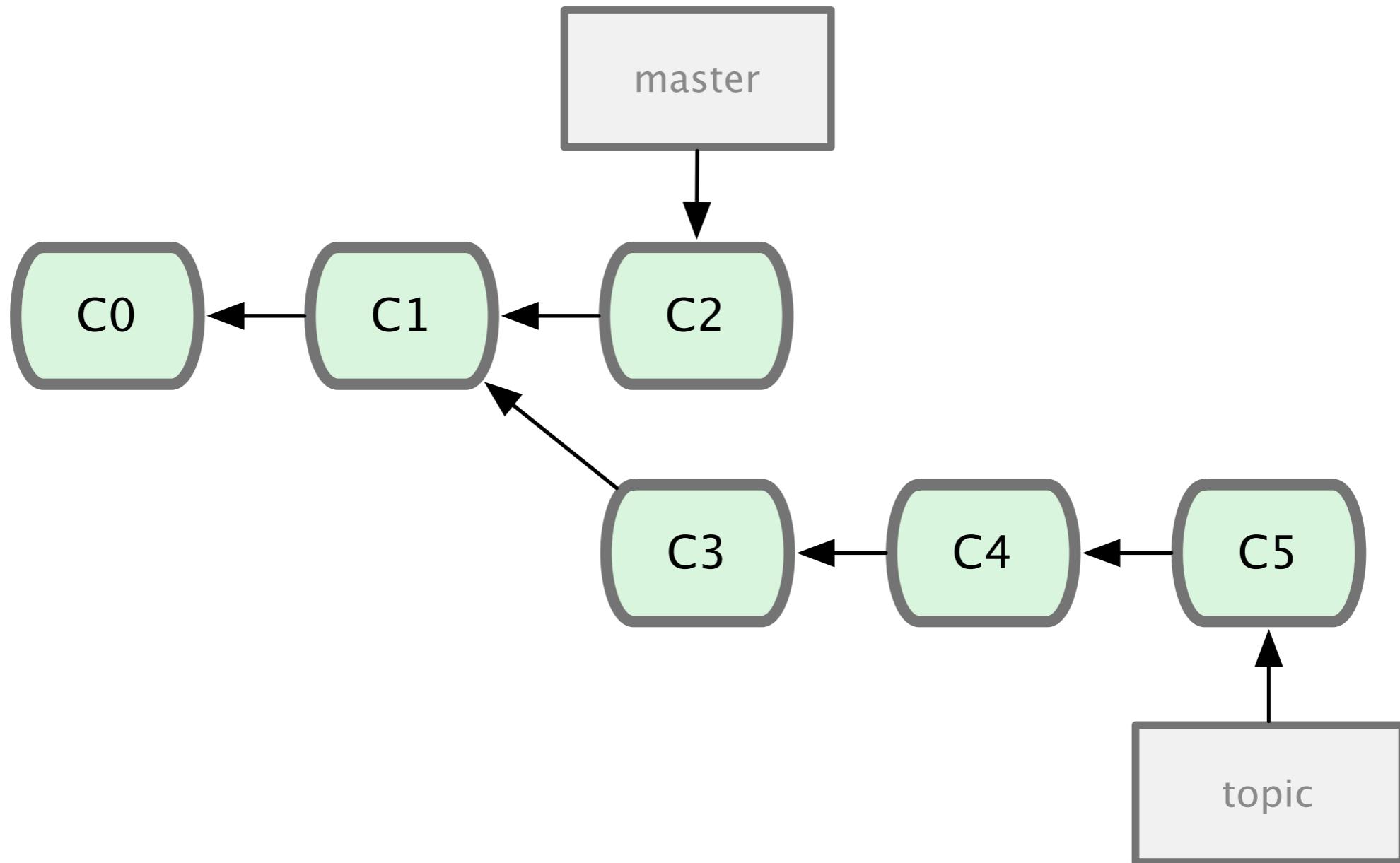


git checkout server
git rebase client

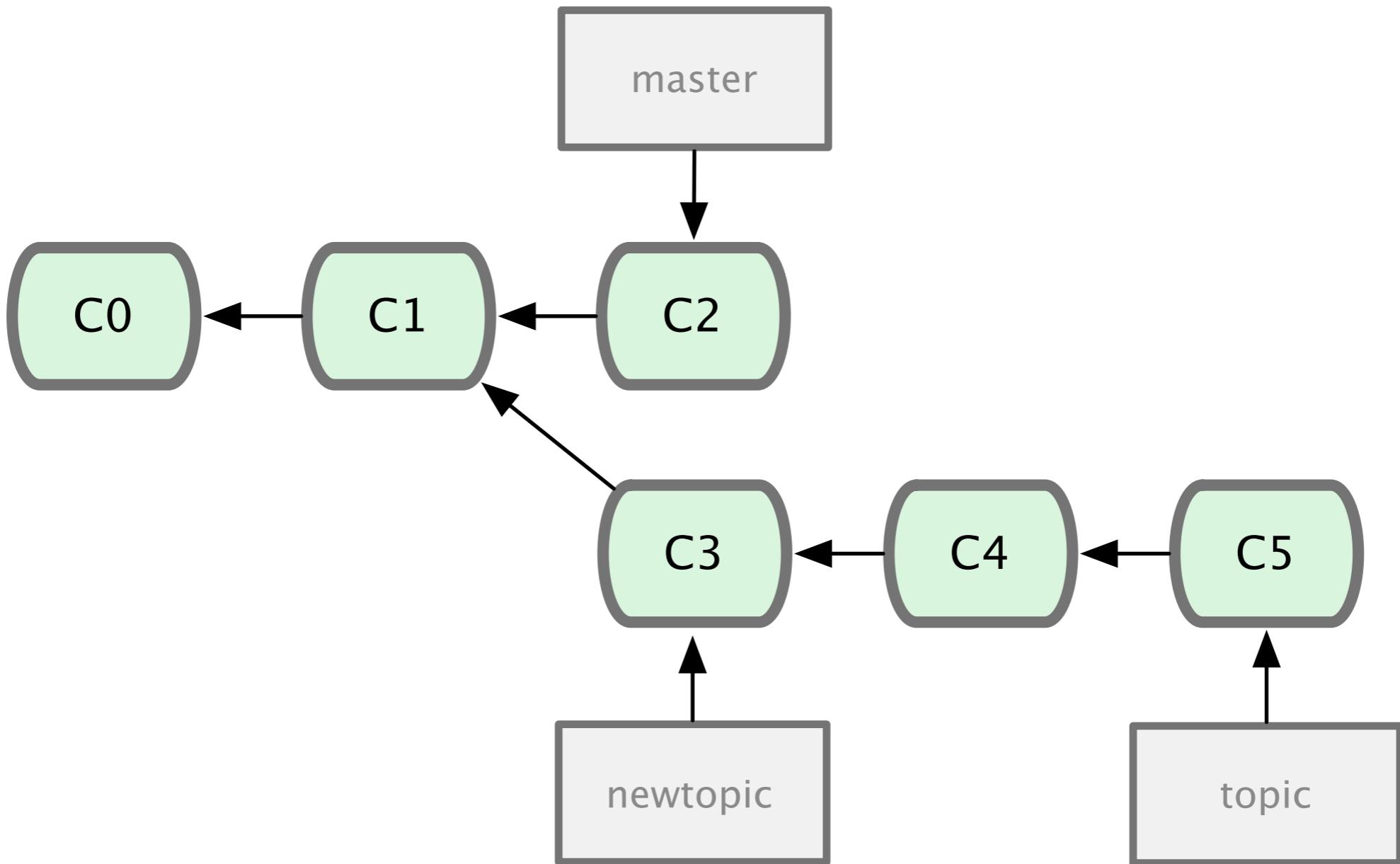
**transplant some of a
topic branch**



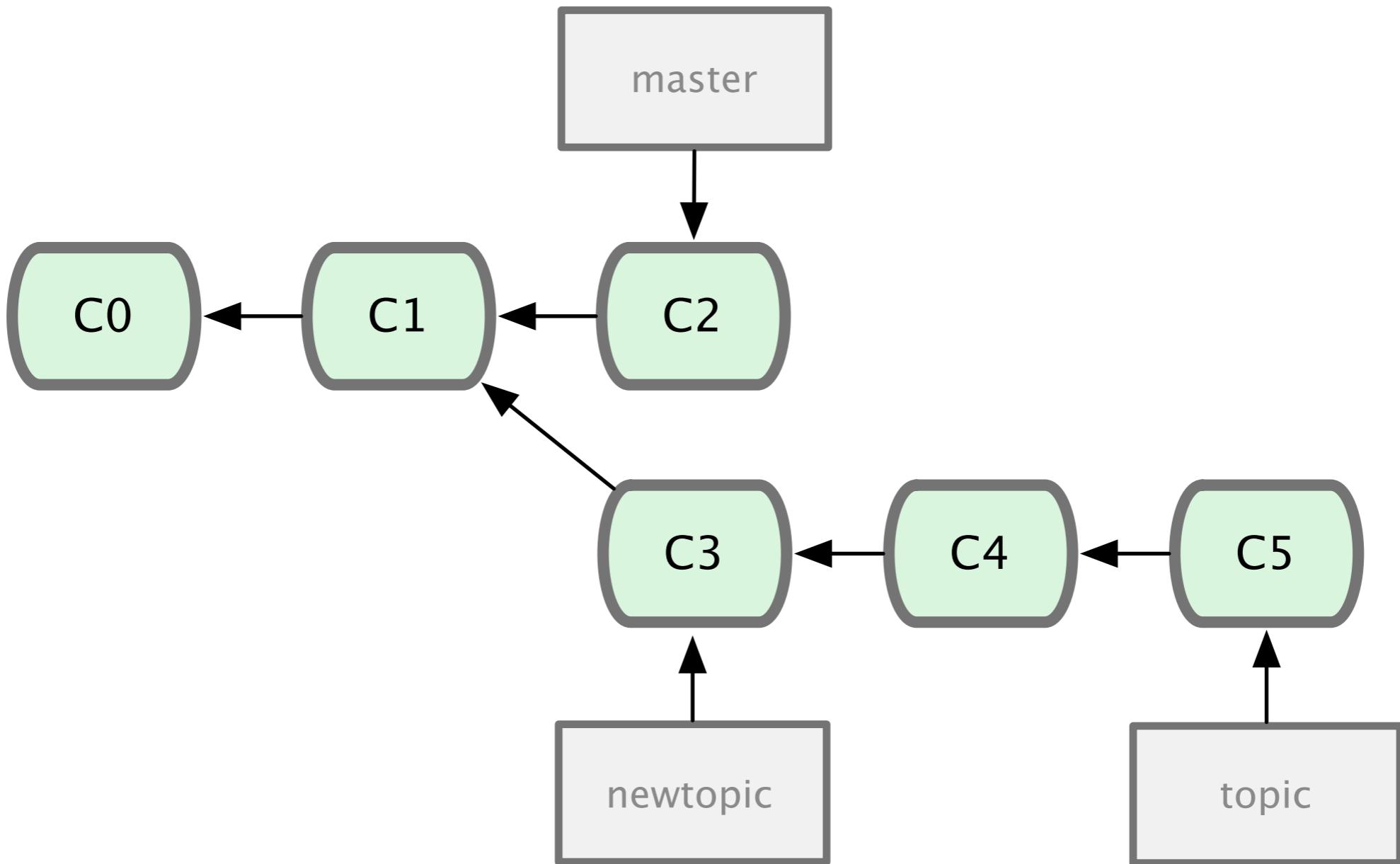




git branch newtopic C3

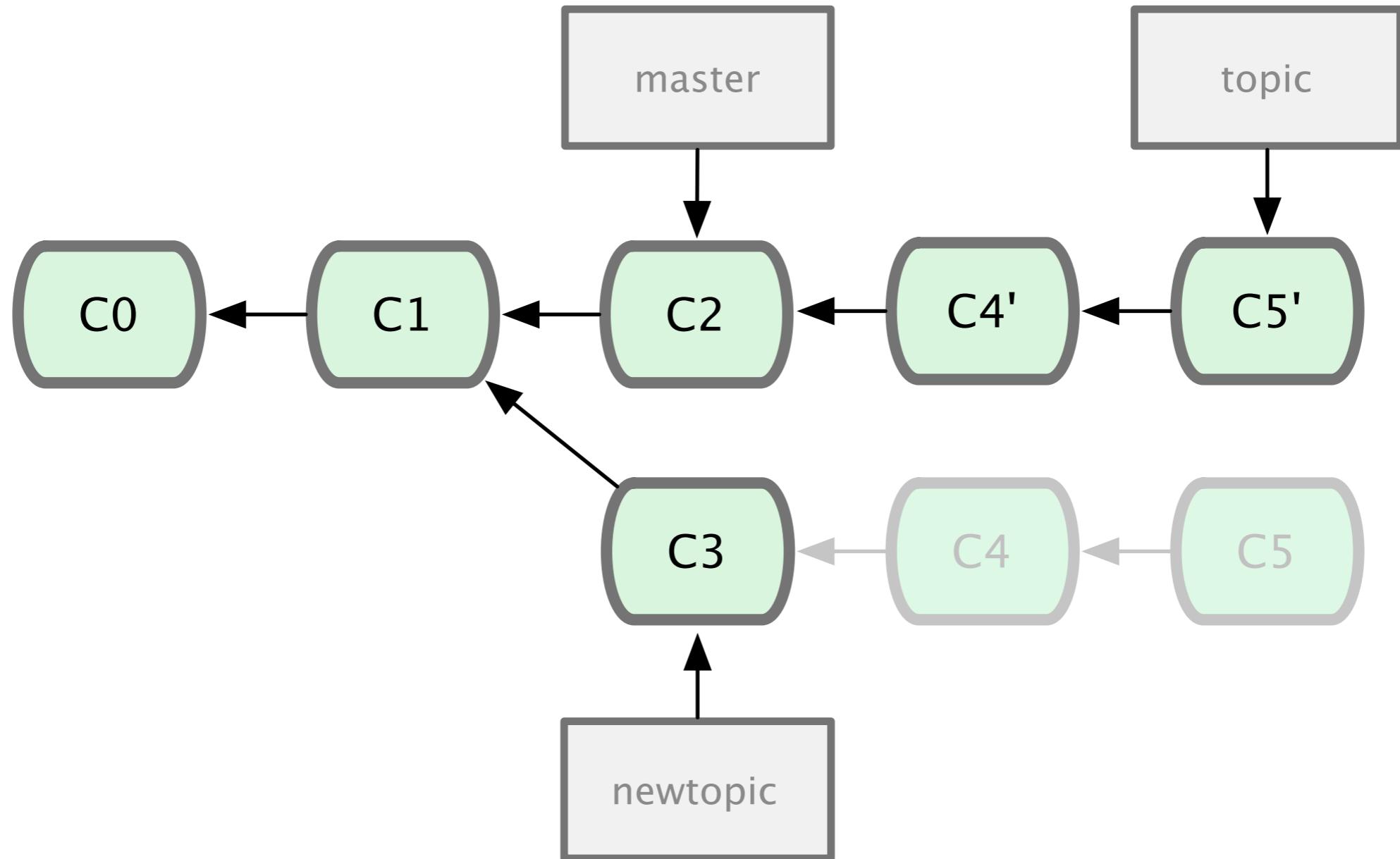


git branch newtopic C3



```
git branch newtopic C3
```

```
git rebase newtopic --onto master
```



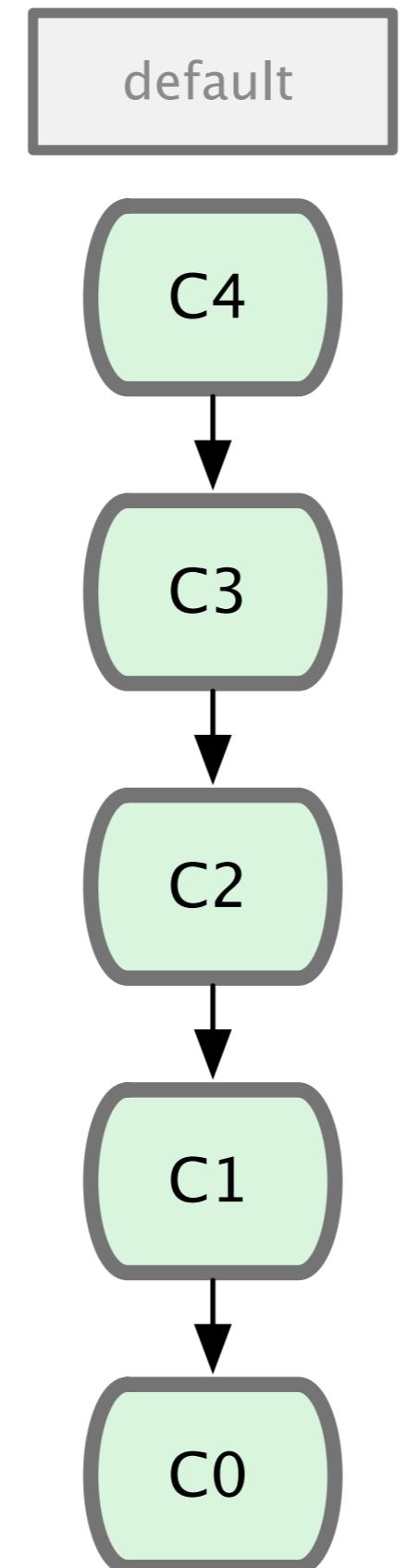
```
git branch newtopic C3
```

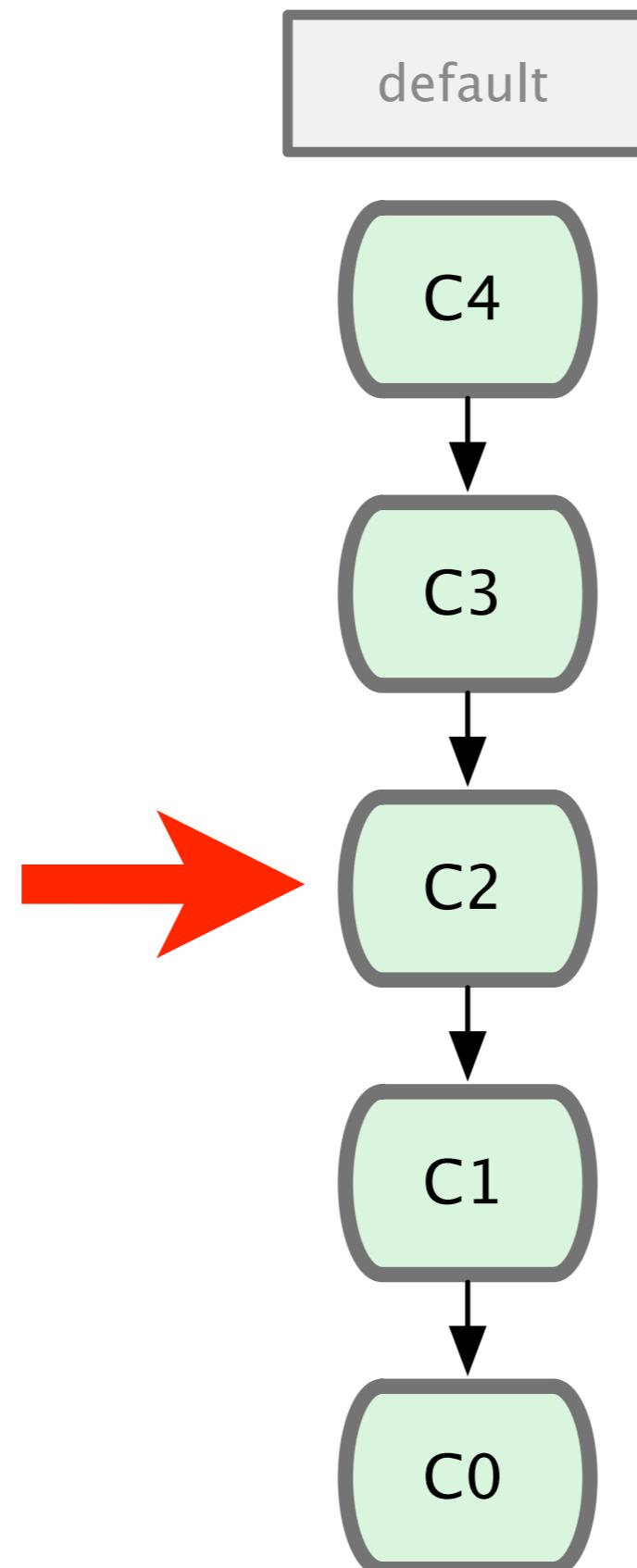
```
git rebase newtopic --onto master
```

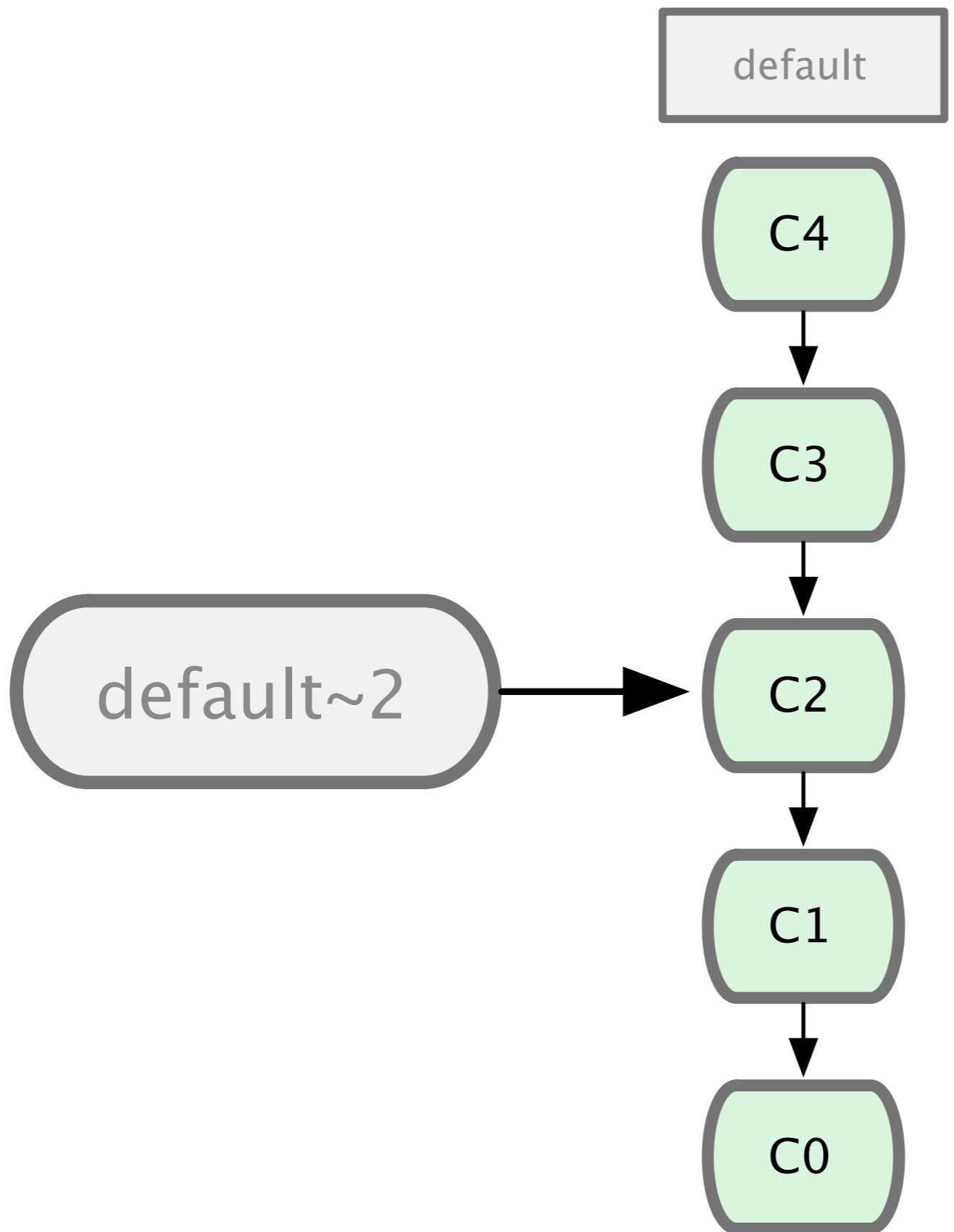
Fixing a commit
several back

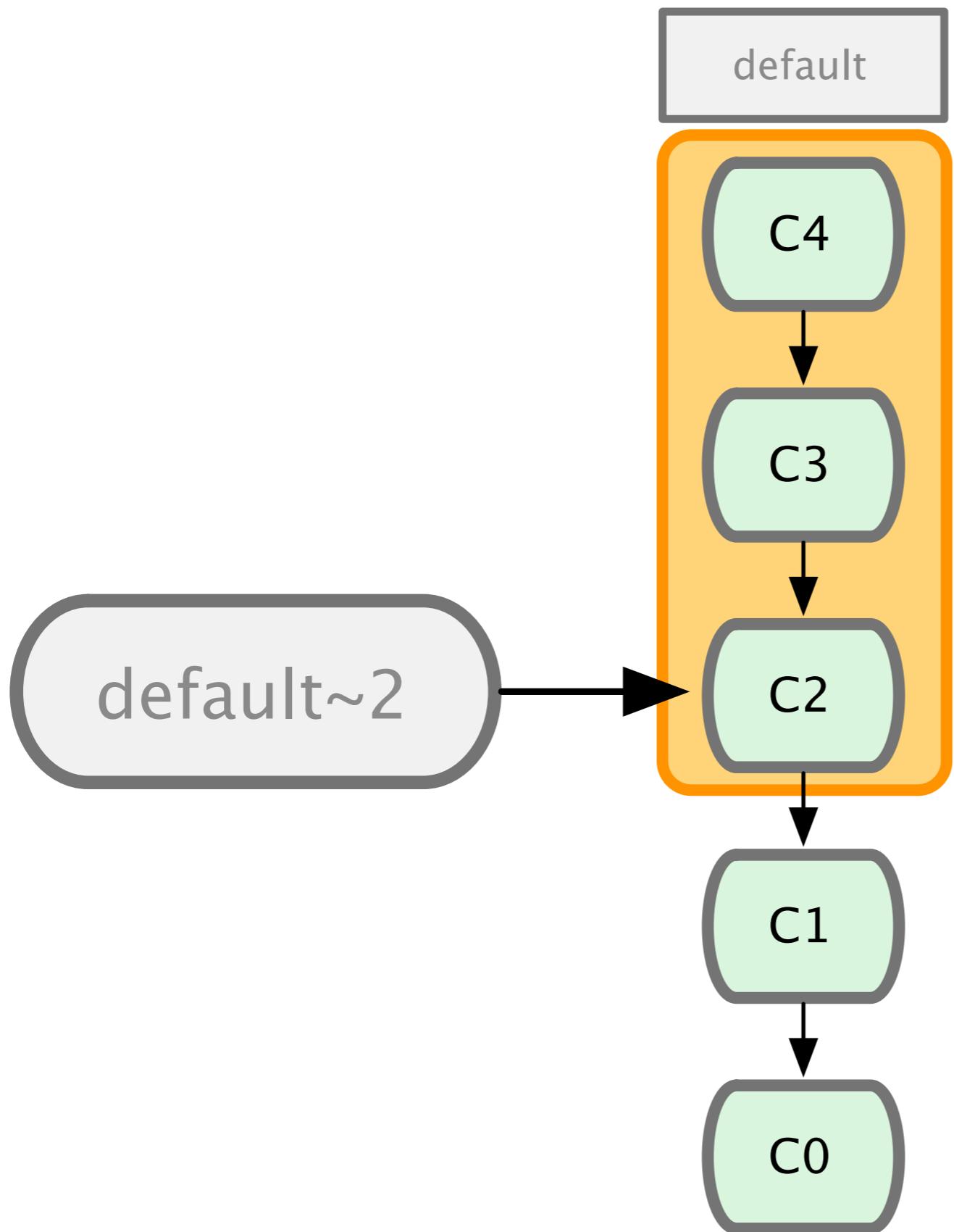
git rebase -i

git rebase --interactive









```
git rebase -i default~2^
```

```
git rebase -i default~2^
```

git rebase -i default~2[^]

```
pick 969c877 git apply --directory broken for new files
pick b75271d git diff <tree>{3,}: do not reverse order of args
pick 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

C2

C3

C4

```
pick 969c877 git apply --directory broken for new files
pick b75271d git diff <tree>{3,}: do not reverse order of args
pick 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

```
pick 969c877 git apply --directory broken for new files
pick b75271d git diff <tree>{3,}: do not reverse order of args
pick 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

```
pick 969c877 git apply --directory broken for new files
pick b75271d git diff <tree>{3,}: do not reverse order of args
pick 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

```
pick 969c877 git apply --directory broken for new files
pick b75271d git diff <tree>{3,}: do not reverse order of args
pick 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

```
pick 969c877 git apply --directory broken for new files
pick b75271d git diff <tree>{3,}: do not reverse order of args
pick 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

```
pick 969c877 git apply --directory broken for new files
pick b75271d git diff <tree>{3,}: do not reverse order of args
pick 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

```
edit 969c877 git apply --directory broken for new files
pick b75271d git diff <tree>{3,}: do not reverse order of args
pick 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

```
$ git rebase -i default~2^
Stopped at 969c877... git apply --directory broken for new files
You can amend the commit now, with
```

```
git commit --amend
```

Once you are satisfied with your changes, run

```
git rebase --continue
```

\$ _

```
$ git rebase -i default~2^  
Stopped at 969c877... git apply --directory broken for new files  
You can amend the commit now, with
```

```
git commit --amend
```

Once you are satisfied with your changes, run

```
git rebase --continue
```

\$ _

edit files

```
$ git rebase -i default~2^  
Stopped at 969c877... git apply --directory broken for new files  
You can amend the commit now, with
```

```
git commit --amend
```

Once you are satisfied with your changes, run

```
git rebase --continue
```

\$ _

edit files
git add

```
$ git rebase -i default~2^  
Stopped at 969c877... git apply --directory broken for new files  
You can amend the commit now, with
```

→ `git commit --amend`

Once you are satisfied with your changes, run

`git rebase --continue`

\$ _

edit files
git add
git commit --amend

```
$ git rebase -i default~2^  
Stopped at 969c877... git apply --directory broken for new files  
You can amend the commit now, with
```

git commit --amend

Once you are satisfied with your changes, run

→ git rebase --continue

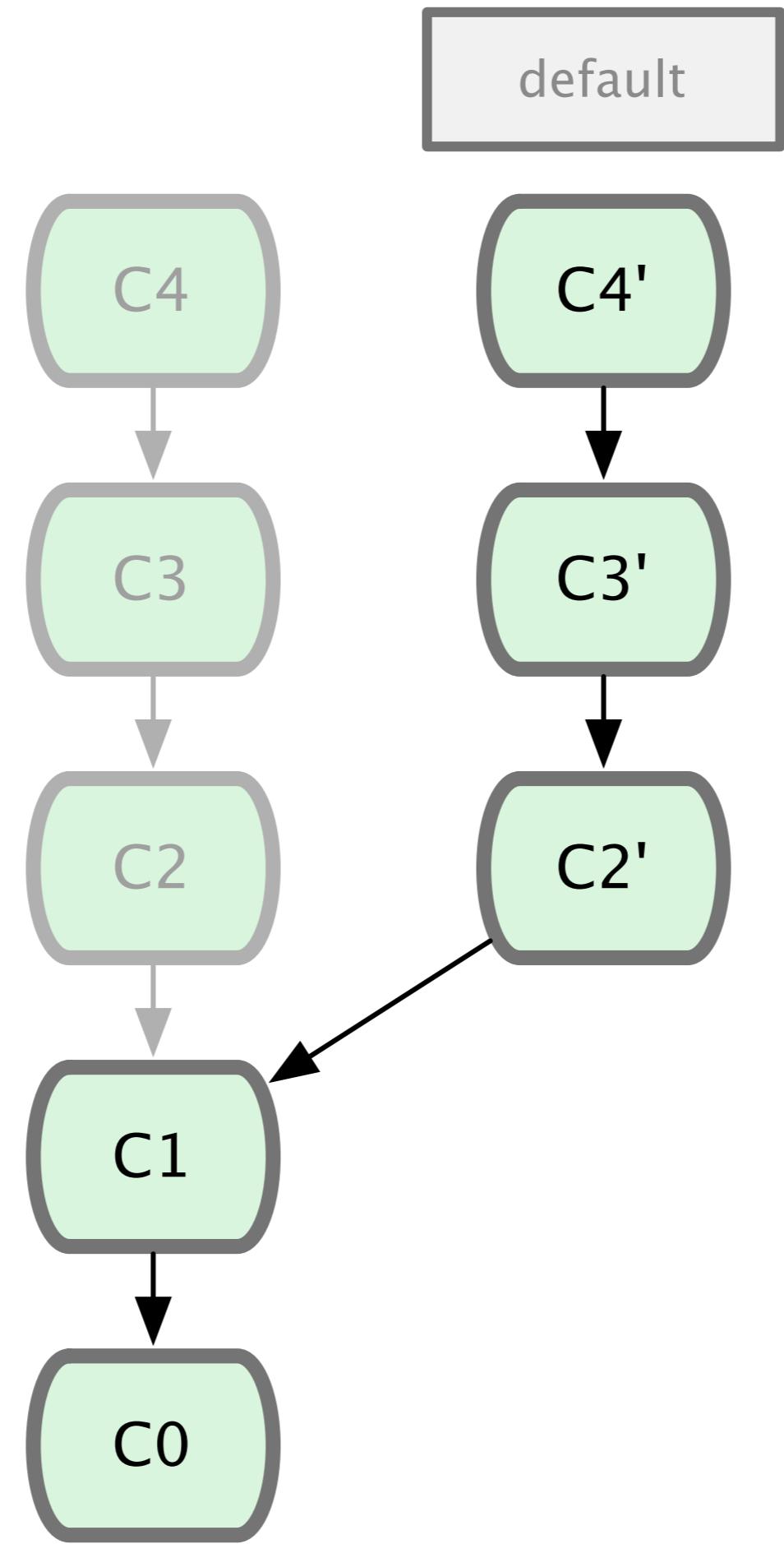
\$ _

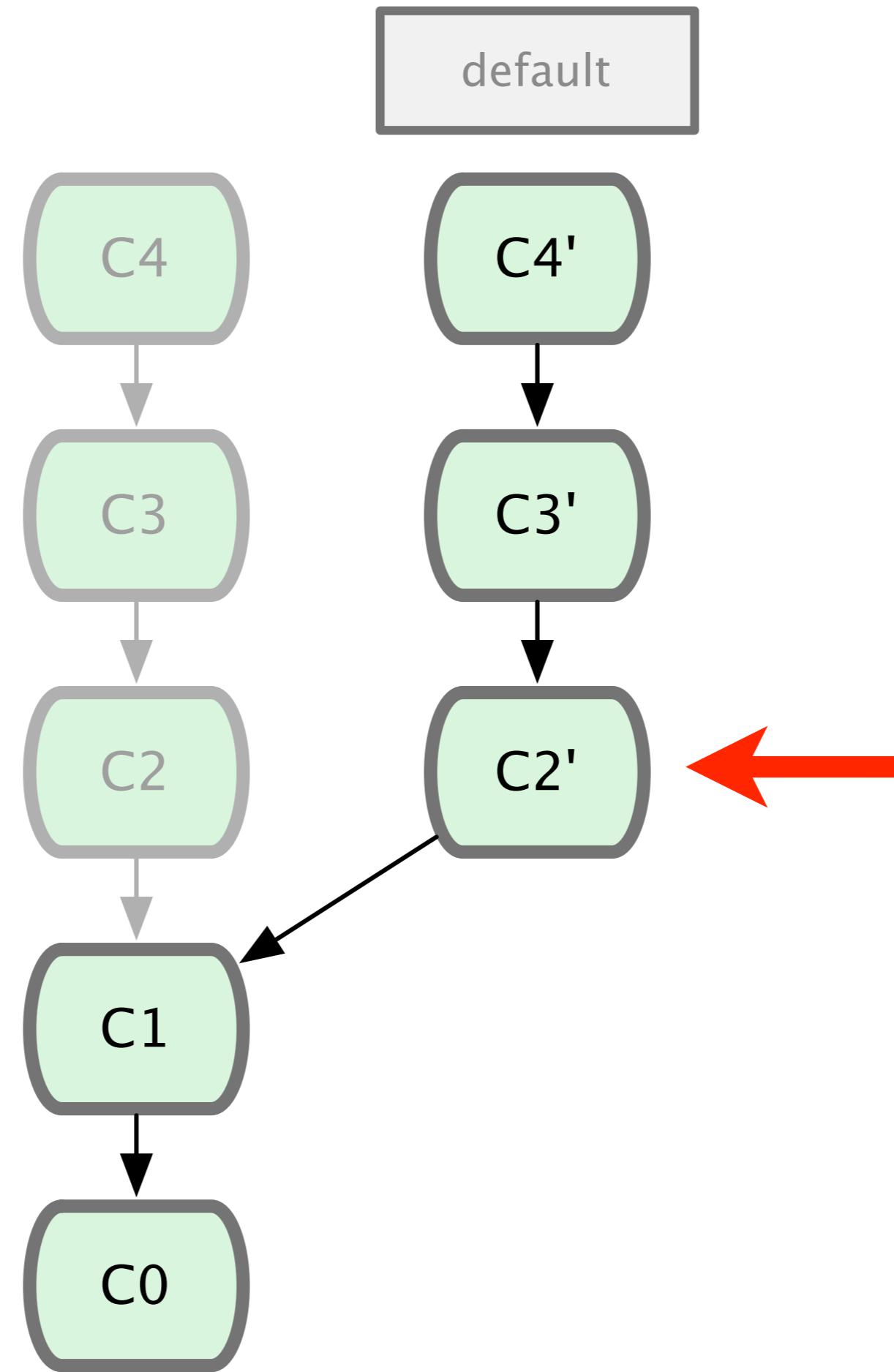
edit files

git add

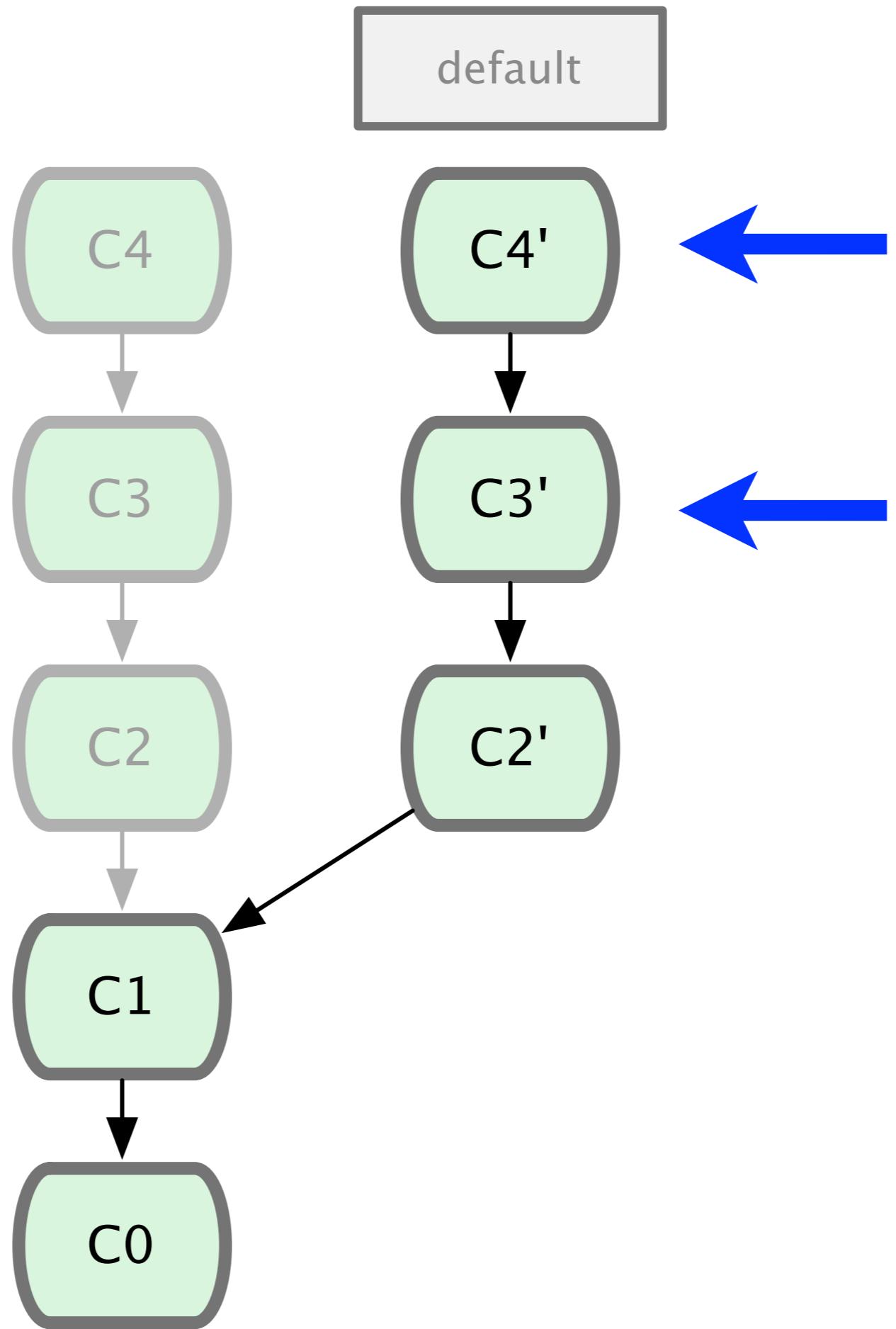
git commit --amend

git rebase --continue





default



Squashing commits
together

```
pick 969c877 git apply --directory broken for new files
pick b75271d git diff <tree>{3,}: do not reverse order of args
pick 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

```
pick 969c877 git apply --directory broken for new files
squash b75271d git diff <tree>{3,}: do not reverse order of args
squash 72d404d test-lib: fix broken printf
```

```
# Rebase f285a2d..5c283eb onto f285a2d
#
# Commands:
# p, pick = use commit
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
#
# If you remove a line here THAT COMMIT WILL BE LOST.
# However, if you remove everything, the rebase will be aborted.
#
~ ~ ~ ~ ~
"~/projects/git/.git/rebase-merge/git-rebase-todo" 14L, 472C
```

```
# This is a combination of 3 commits.
# The first commit's message is:
git apply --directory broken for new files

# This is the 2nd commit message:

git diff <tree>{3,}: do not reverse order of args

# This is the 3rd commit message:

test-lib: fix broken printf

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Author:    Jeff King <peff@peff.net>
#
# Not currently on any branch.
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:    builtin-apply.c
# modified:    builtin-diff.c
# modified:    t/t4013-diff-various.sh
# new file:    t/t4013/diff.diff_master_master^_side
# modified:    t/t4128-apply-root.sh
# modified:    t/test-lib.sh
#
~
~

".git/COMMIT_EDITMSG" 39L, 1454C
```

```
# This is a combination of 3 commits.
# The first commit's message is:
git apply --directory broken for new files

# This is the 2nd commit message:

git diff <tree>{3,}: do not reverse order of args

# This is the 3rd commit message:

test-lib: fix broken printf

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Author:    Jeff King <peff@peff.net>
#
# Not currently on any branch.
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:    builtin-apply.c
# modified:    builtin-diff.c
# modified:    t/t4013-diff-various.sh
# new file:    t/t4013/diff.diff_master_master^_side
# modified:    t/t4128-apply-root.sh
# modified:    t/test-lib.sh
#
~
~

".git/COMMIT_EDITMSG" 39L, 1454C
```

```
# This is a combination of 3 commits.
# The first commit's message is:
git apply --directory broken for new files

# This is the 2nd commit message:

git diff <tree>{3,}: do not reverse order of args

# This is the 3rd commit message:

test-lib: fix broken printf

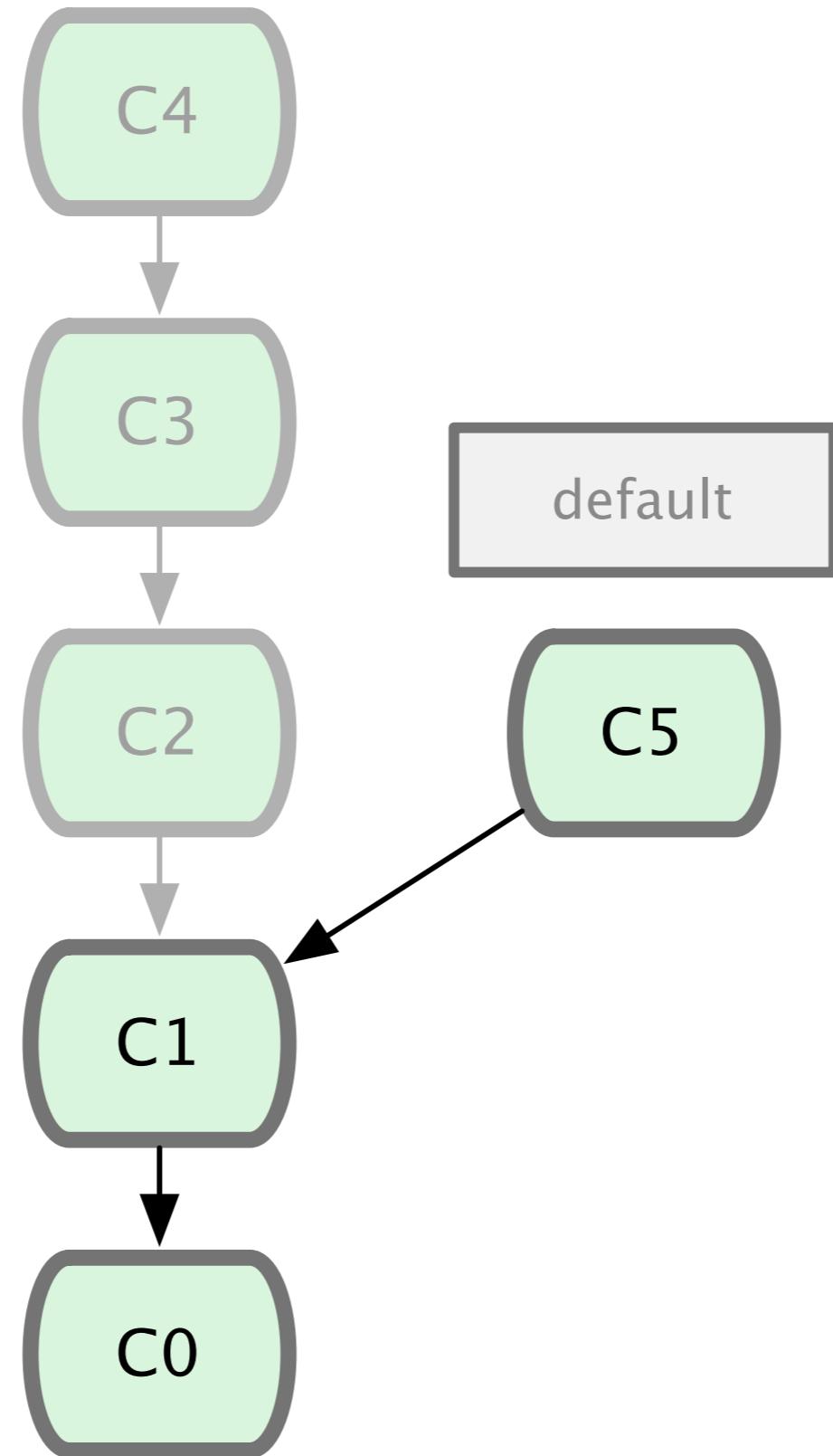
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Author:    Jeff King <peff@peff.net>
#
# Not currently on any branch.
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
# modified:    builtin-apply.c
# modified:    builtin-diff.c
# modified:    t/t4013-diff-various.sh
# new file:    t/t4013/diff.diff_master_master^_side
# modified:    t/t4128-apply-root.sh
# modified:    t/test-lib.sh
#
~
~

".git/COMMIT_EDITMSG" 39L, 1454C
```

```
# This is a combination of 3 commits.  
# The first commit's message is:  
git apply --directory broken for new files  
  
# This is the 2nd commit message:  
  
git diff <tree>{3,}: do not reverse order of args
```

```
# This is the 3rd commit message:  
  
test-lib: fix broken printf
```

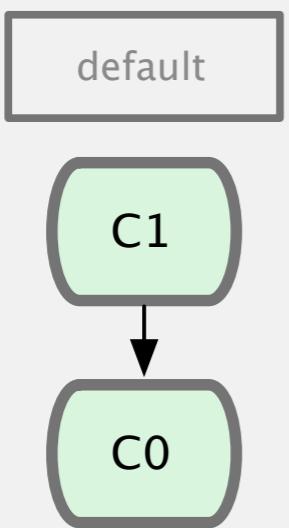
```
# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the commit.  
#  
# Author: Jeff King <peff@peff.net>  
#  
# Not currently on any branch.  
# Changes to be committed:  
#   (use "git reset HEAD <file>..." to unstage)  
#  
# modified: builtin-apply.c  
# modified: builtin-diff.c  
# modified: t/t4013-diff-various.sh  
# new file: t/t4013/diff.diff_master_master^_side  
# modified: t/t4128-apply-root.sh  
# modified: t/test-lib.sh  
#  
~  
~  
.git/COMMIT_EDITMSG" 39L, 1454C
```



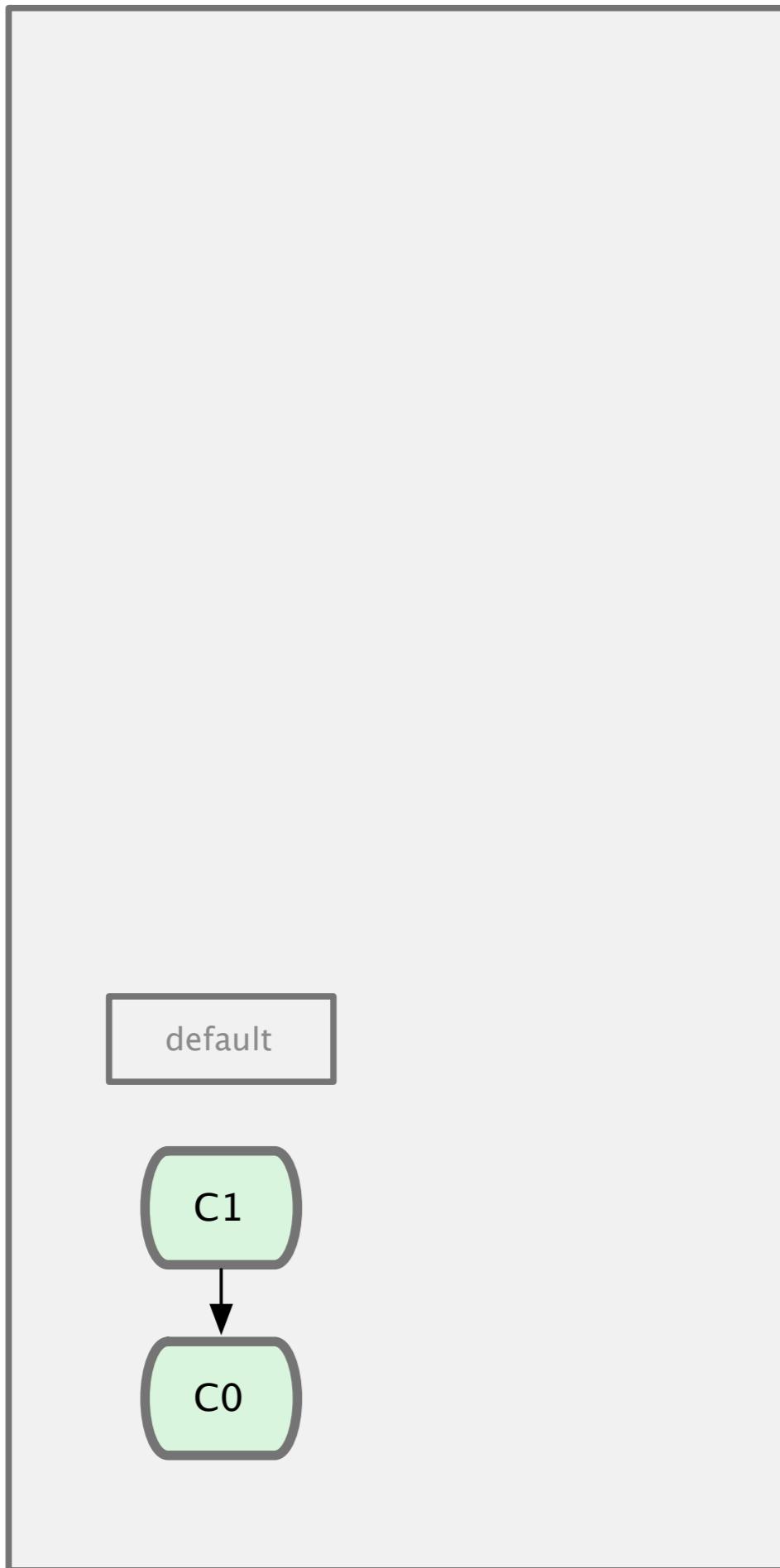
The Perils

scott

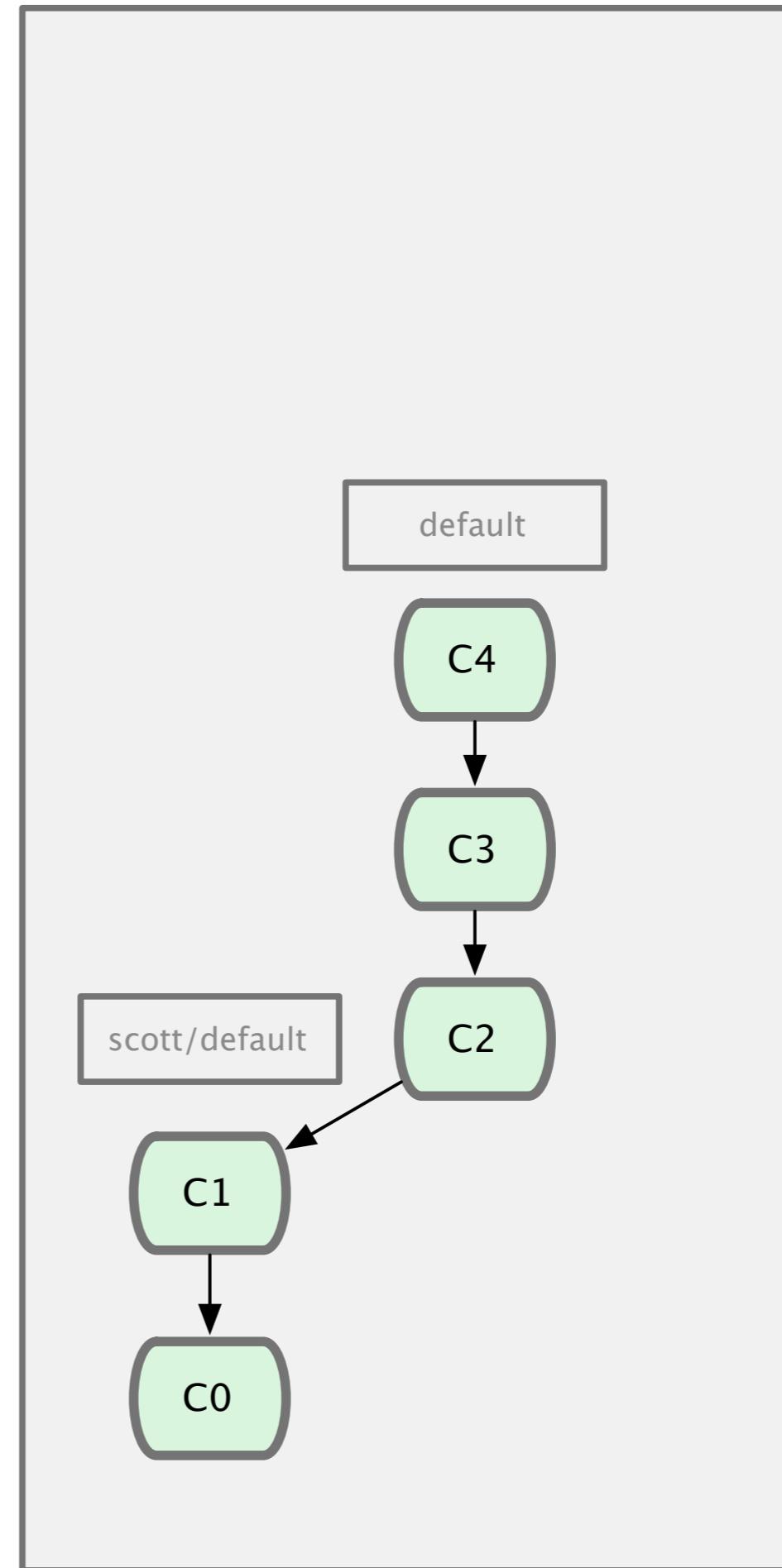
jessica



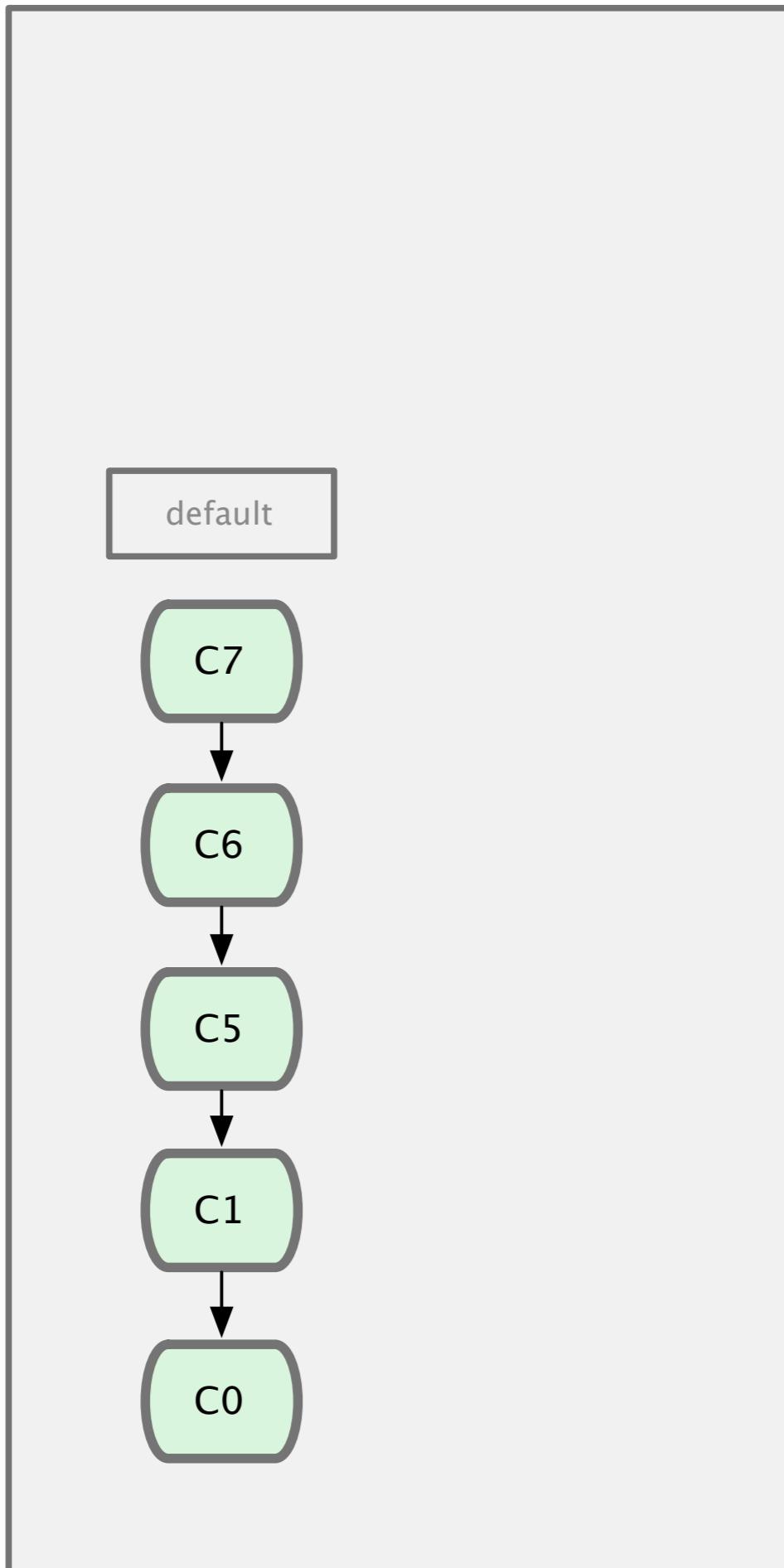
scott



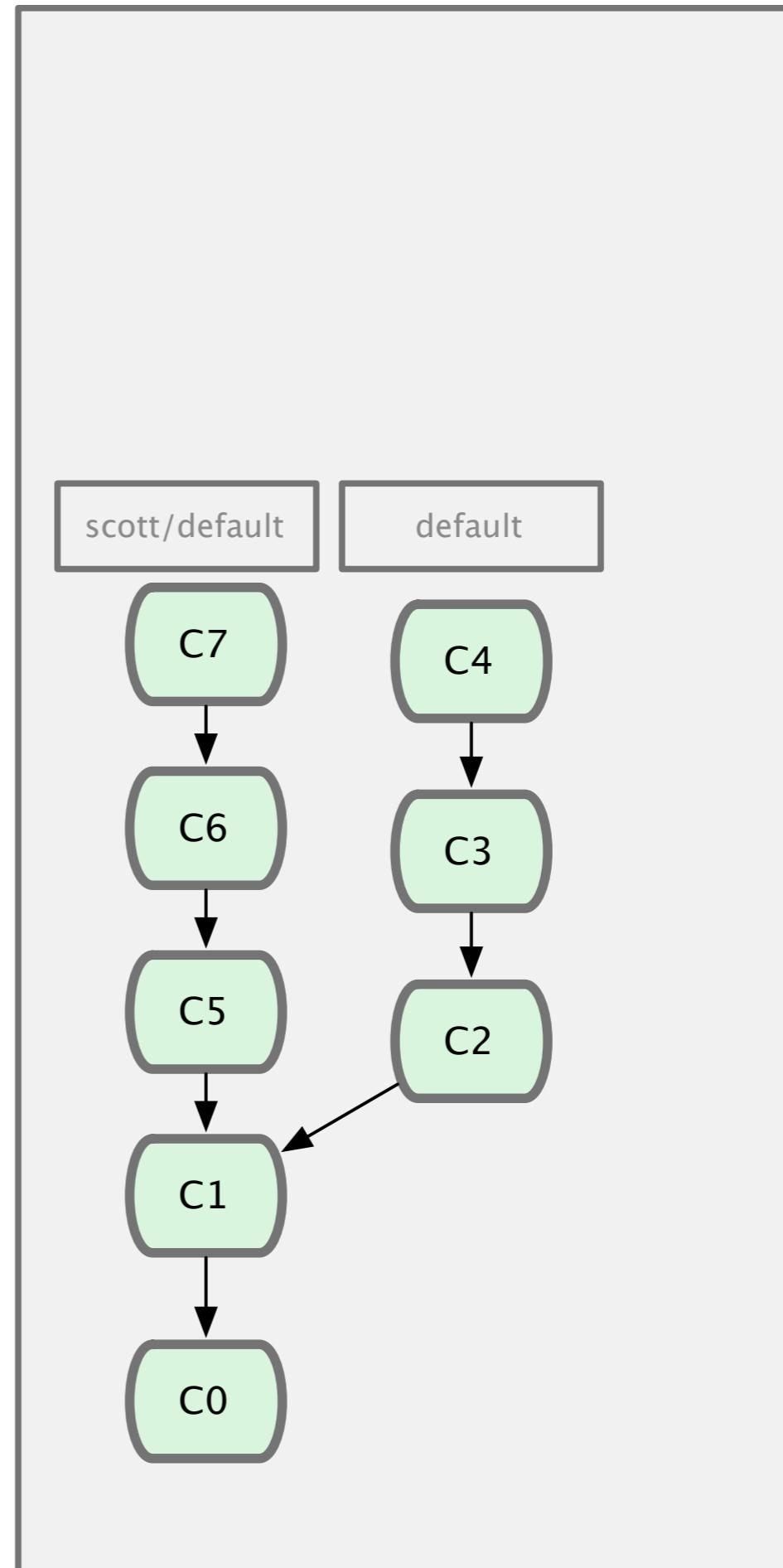
jessica



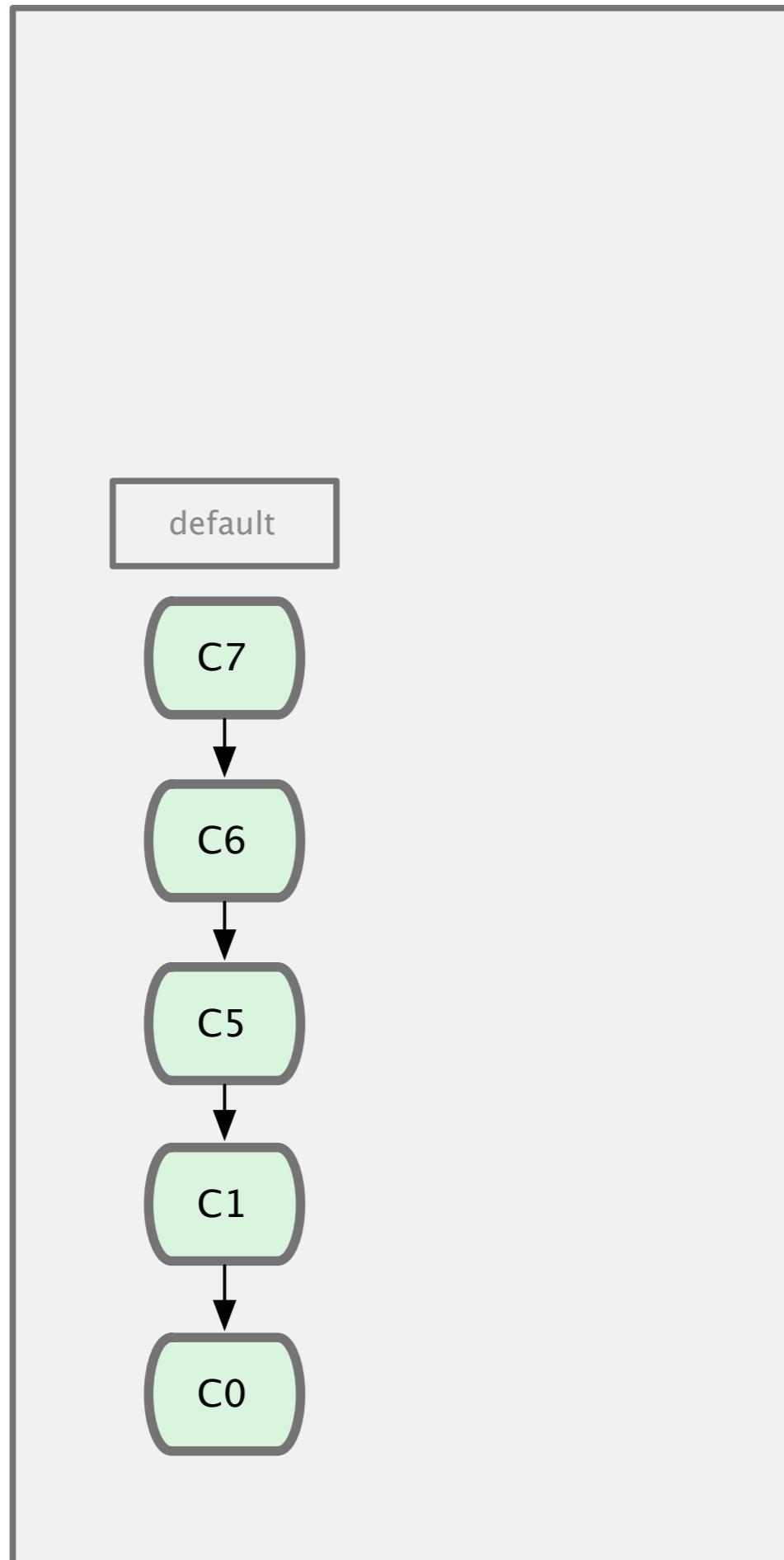
scott



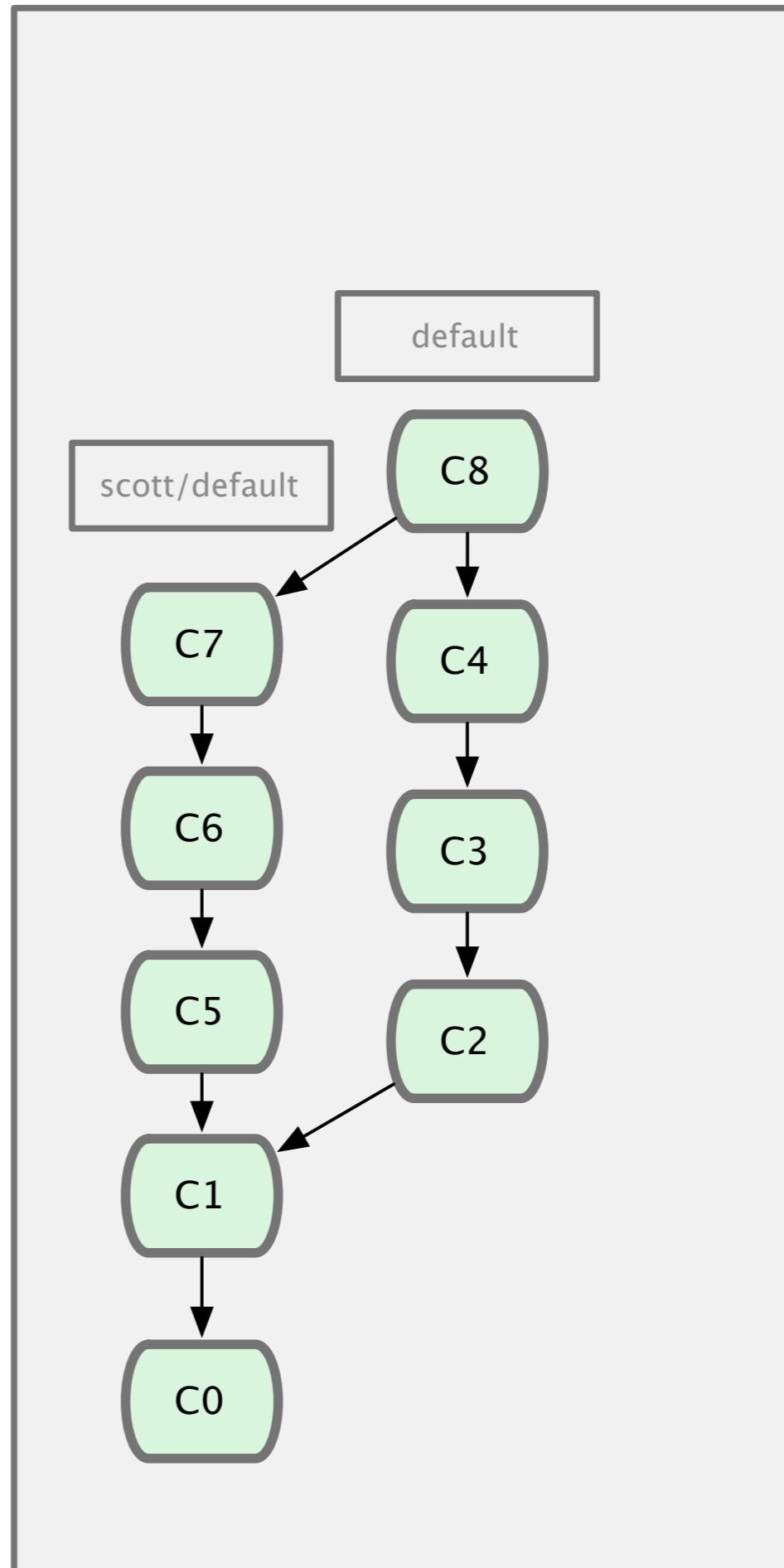
jessica



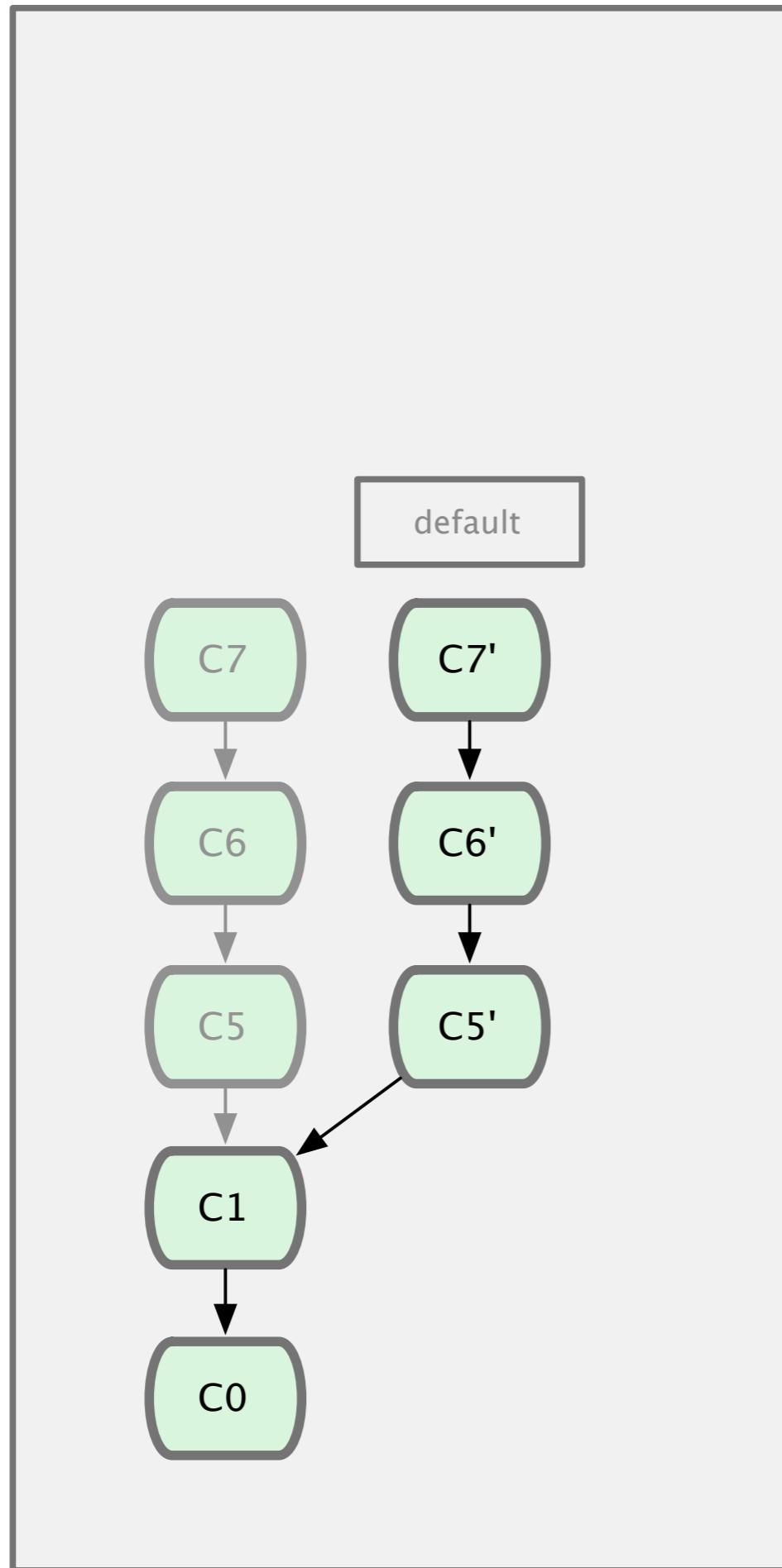
scott



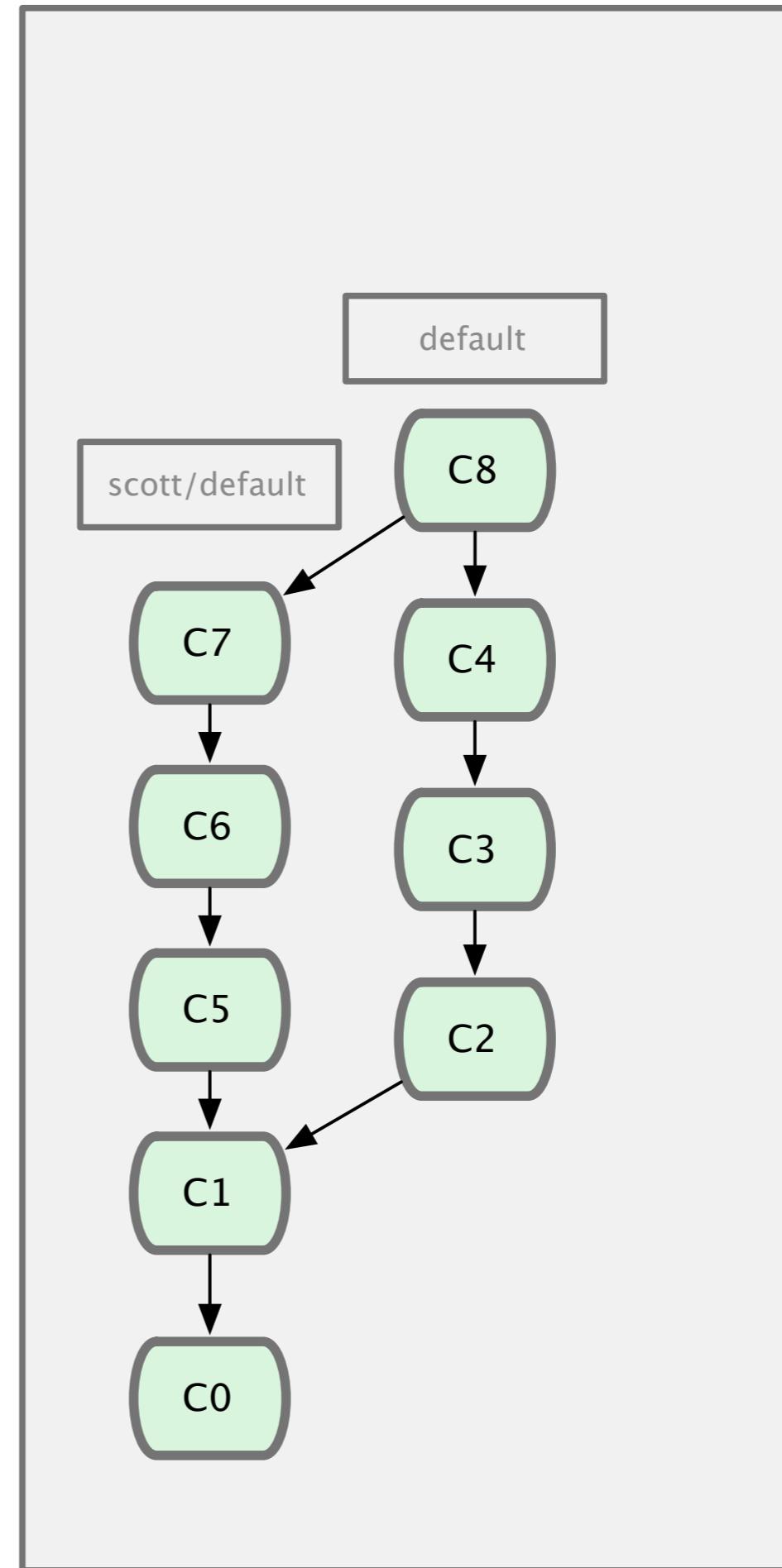
jessica



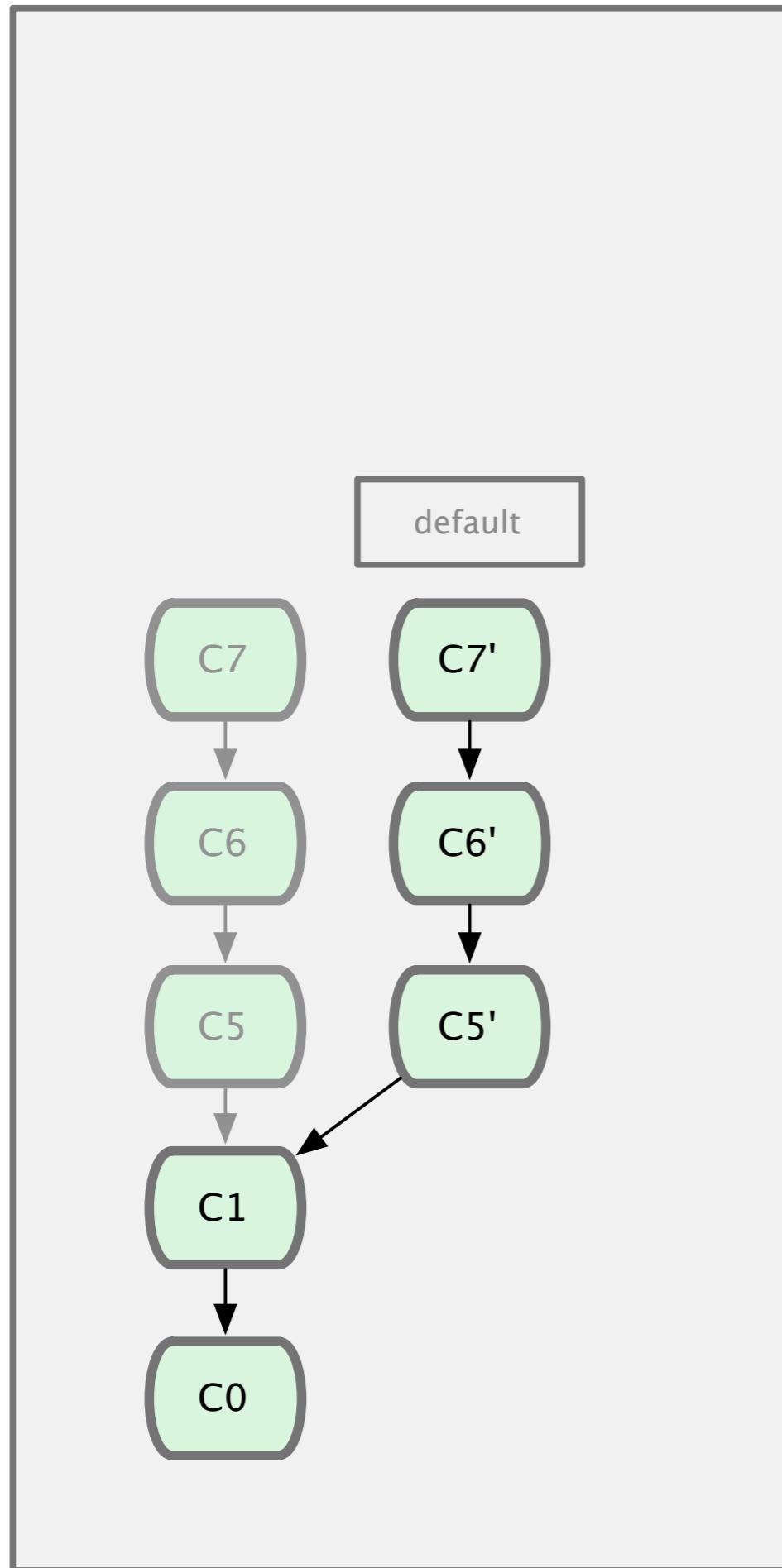
scott



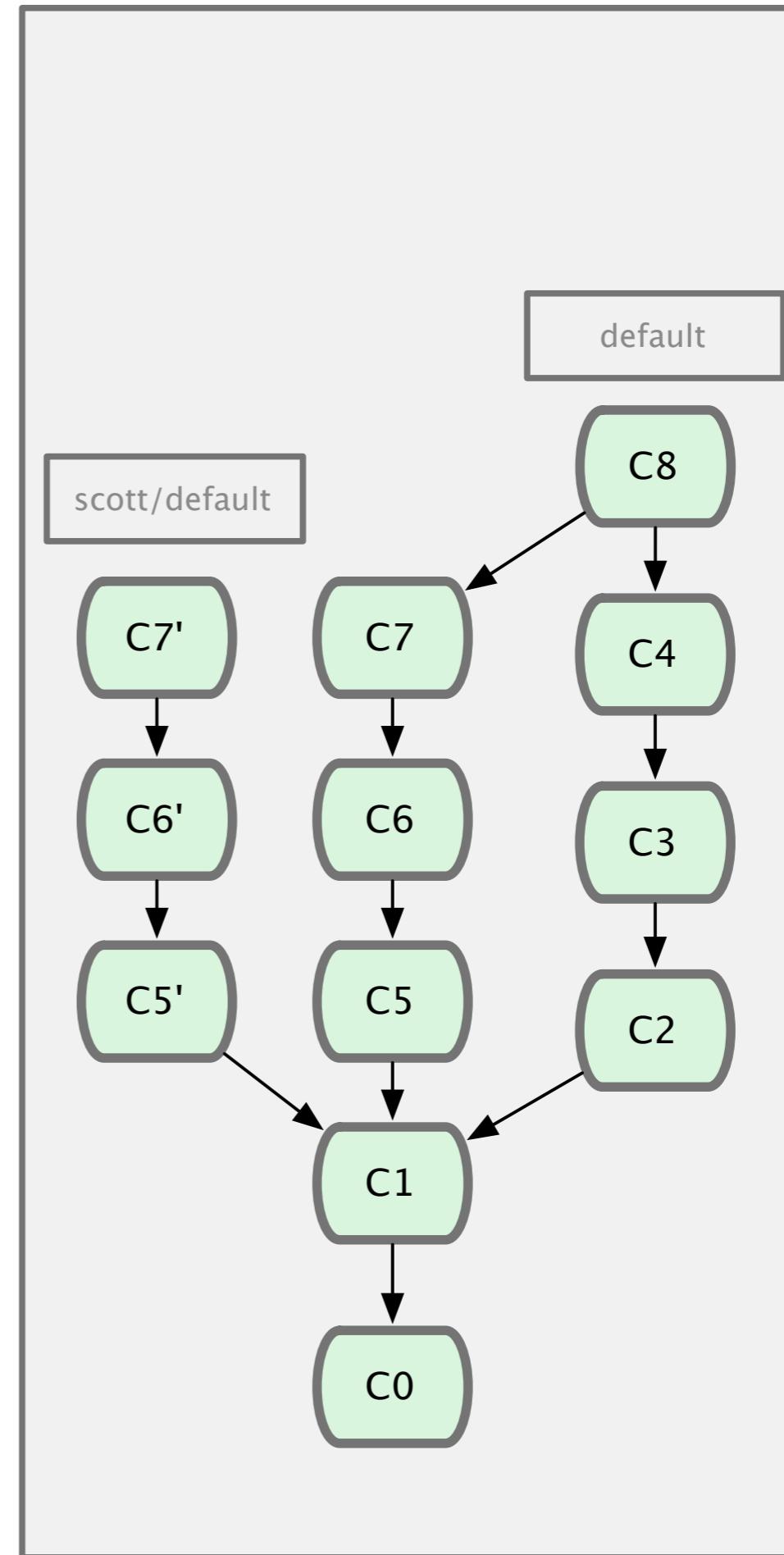
jessica



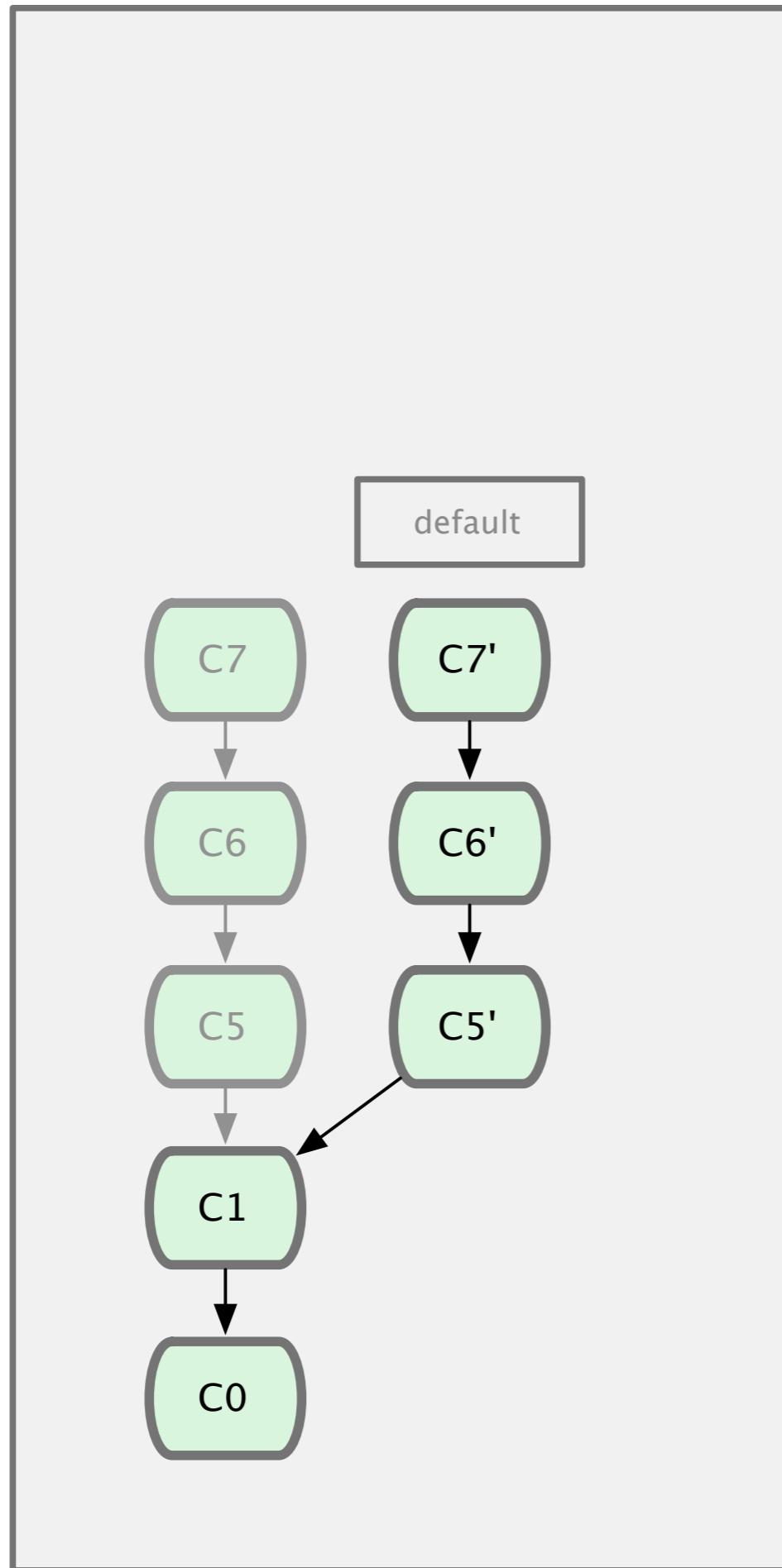
scott



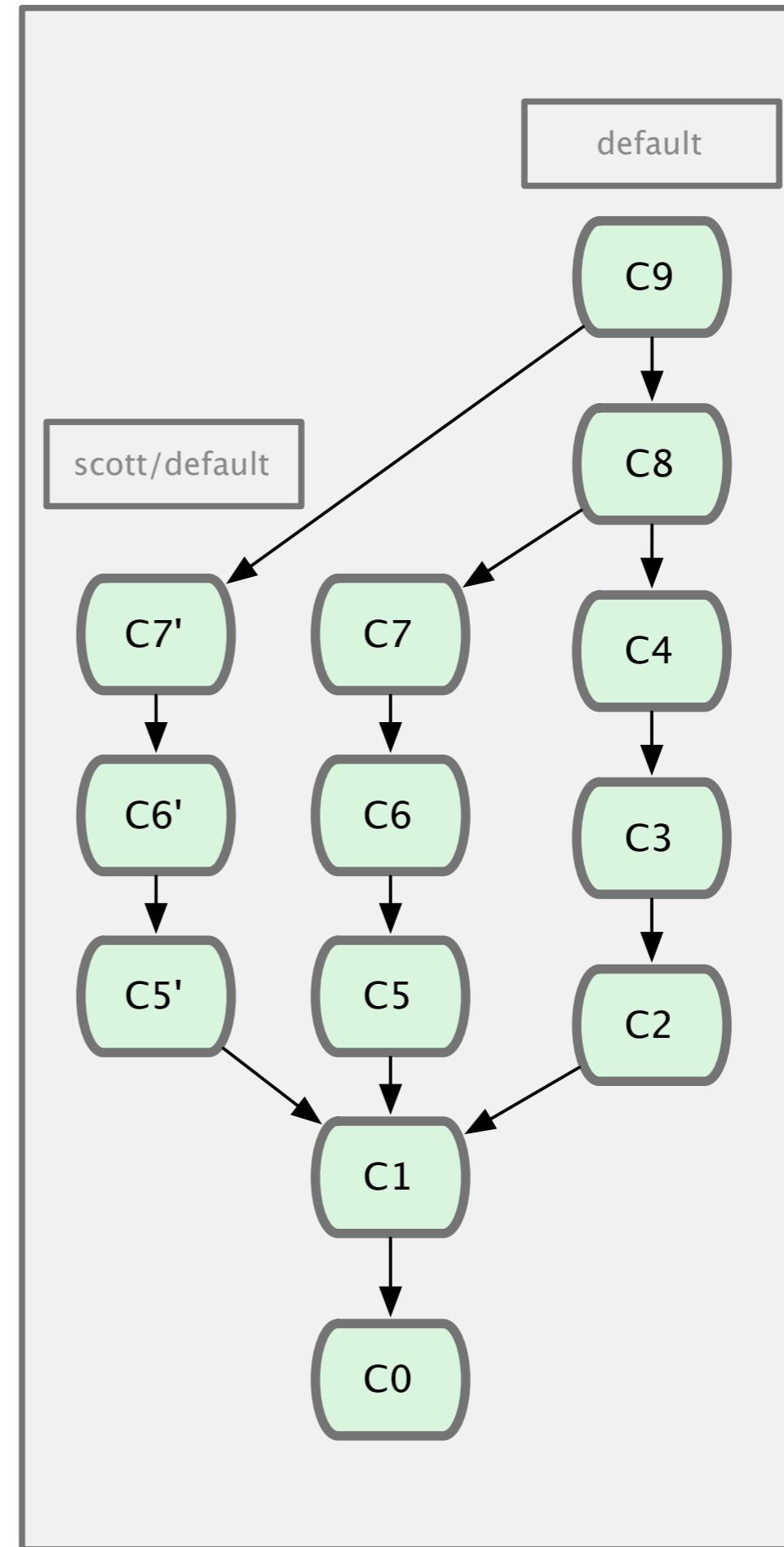
jessica



scott



jessica



Filter Branch

History Revision on Steroids

remove all instances of a
file from every commit

```
git filter-branch  
--tree-filter 'rm -f filename'  
HEAD
```

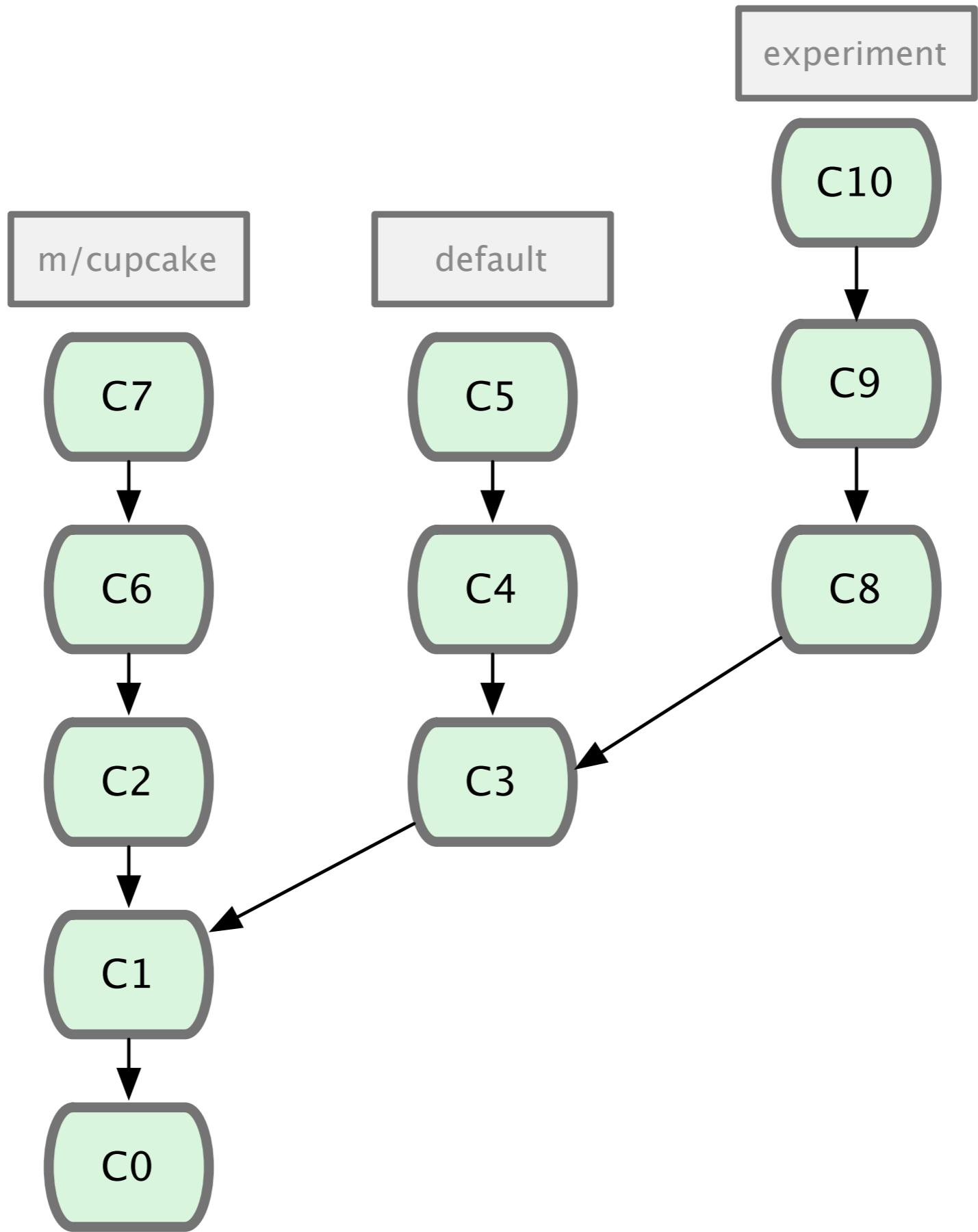
change your email
in all commits

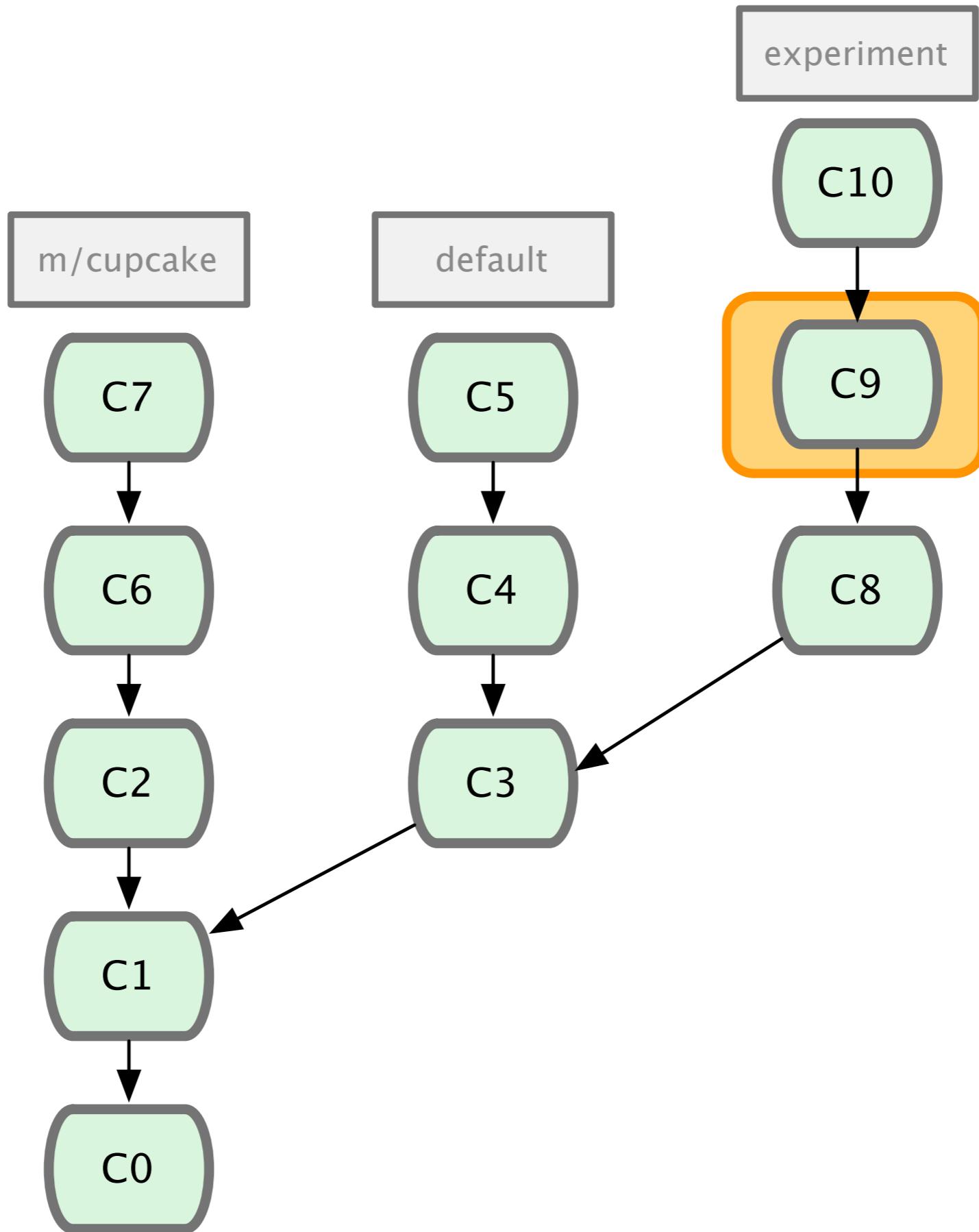
```
git filter-branch --env-filter  
"export GIT_AUTHOR_EMAIL=you@email.com"  
HEAD
```

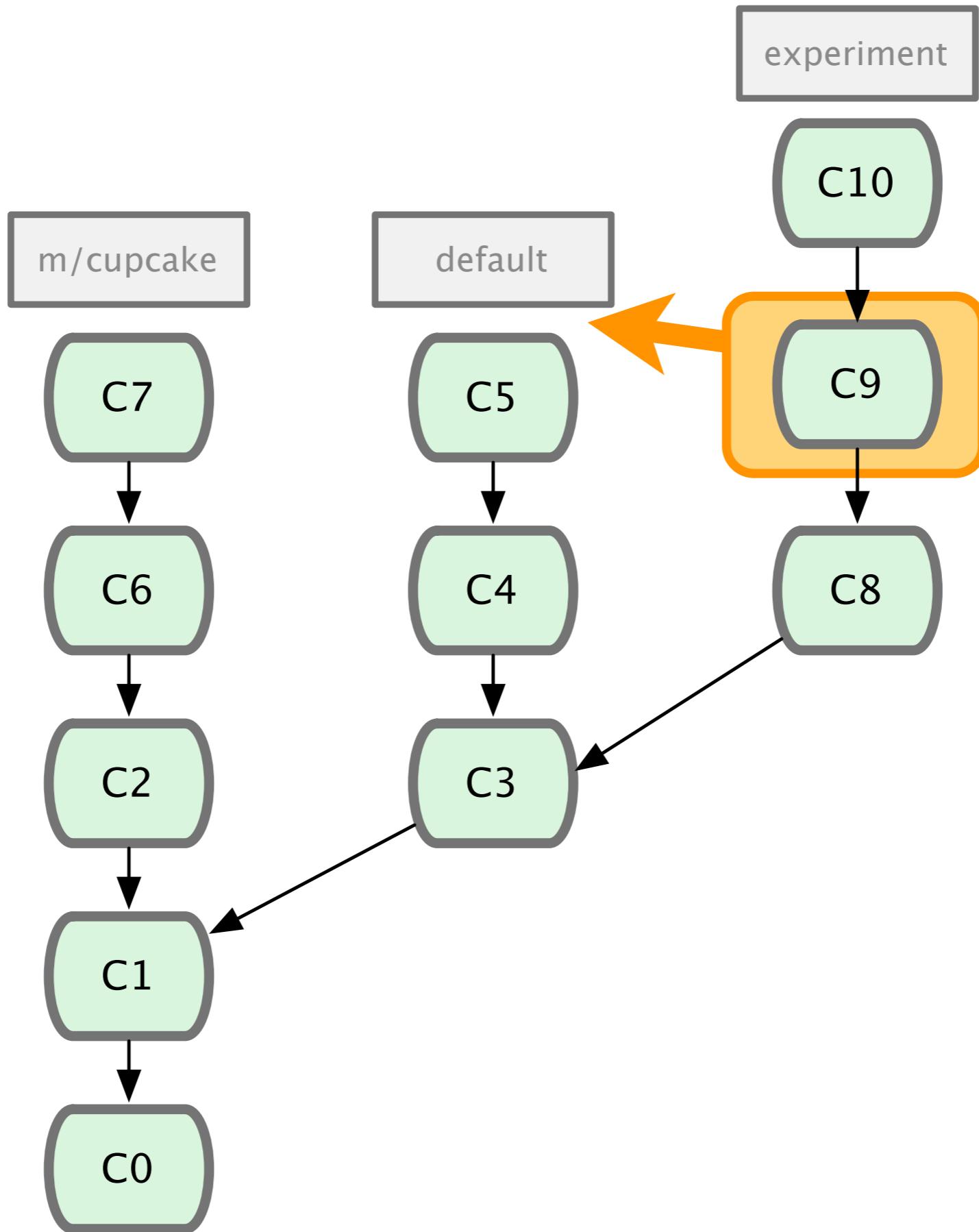
```
git filter-branch --env-filter  
"export GIT_AUTHOR_EMAIL=you@email.com"  
origin/master..HEAD
```

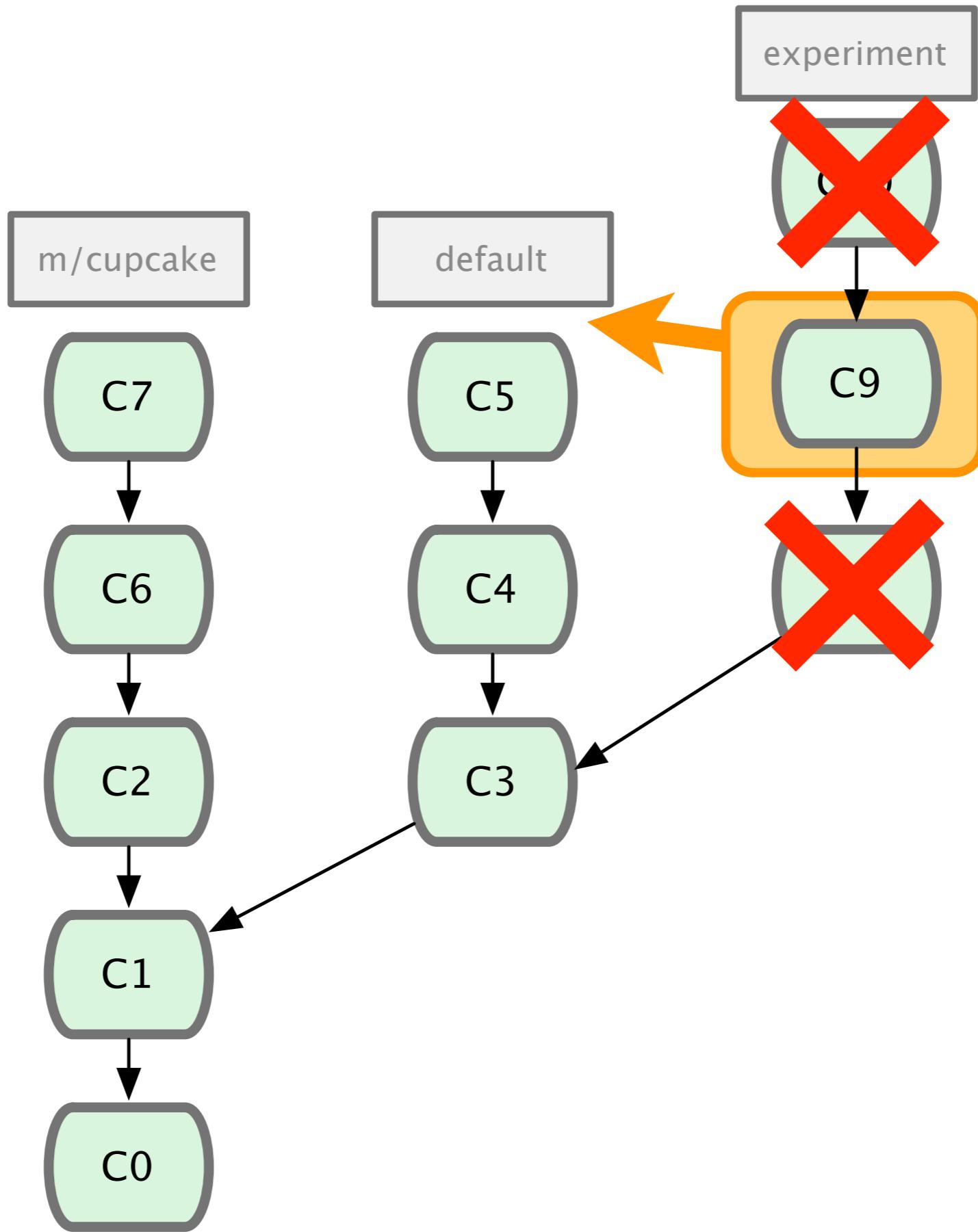
Cherry Picking

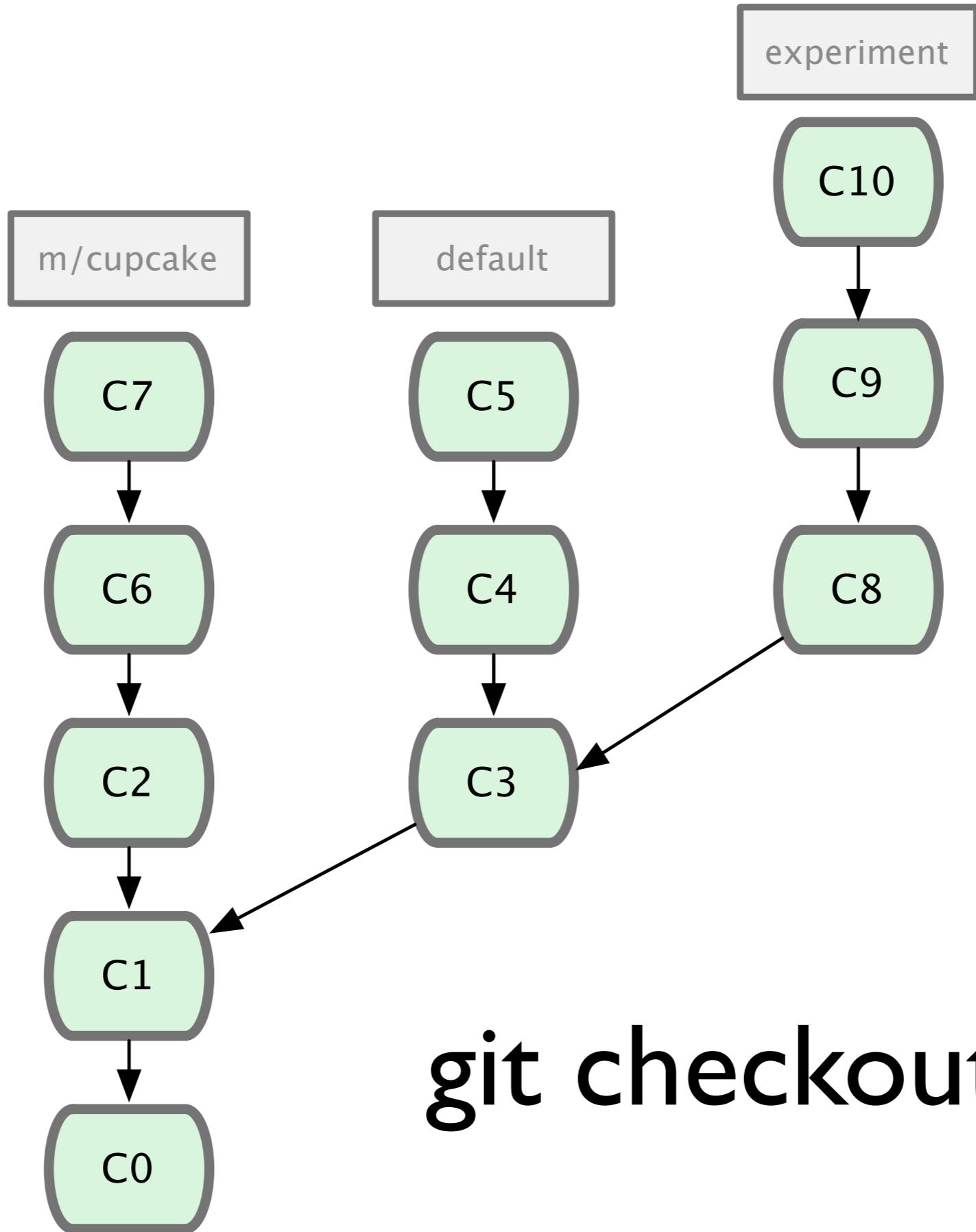
git cherry-pick 3f40a



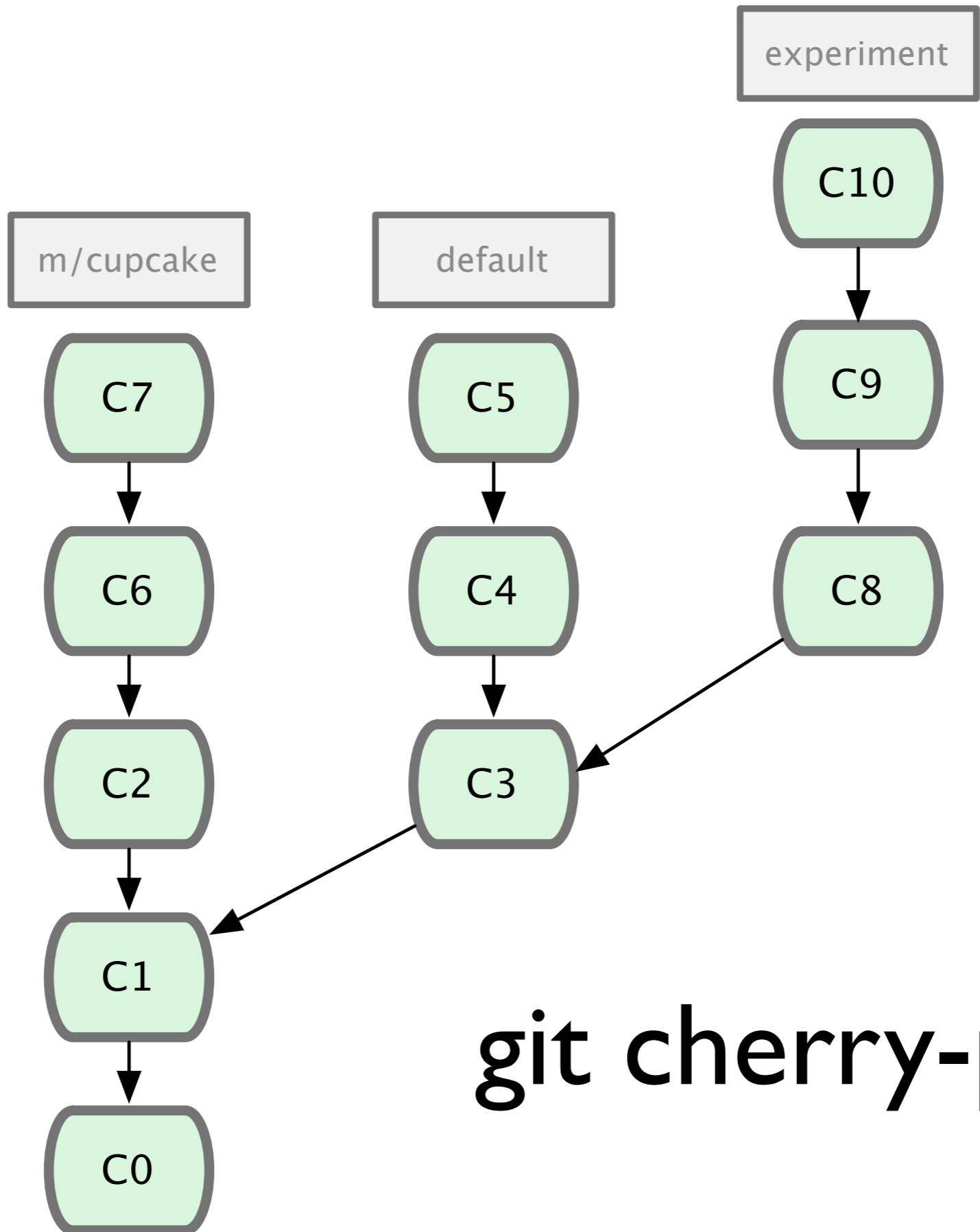




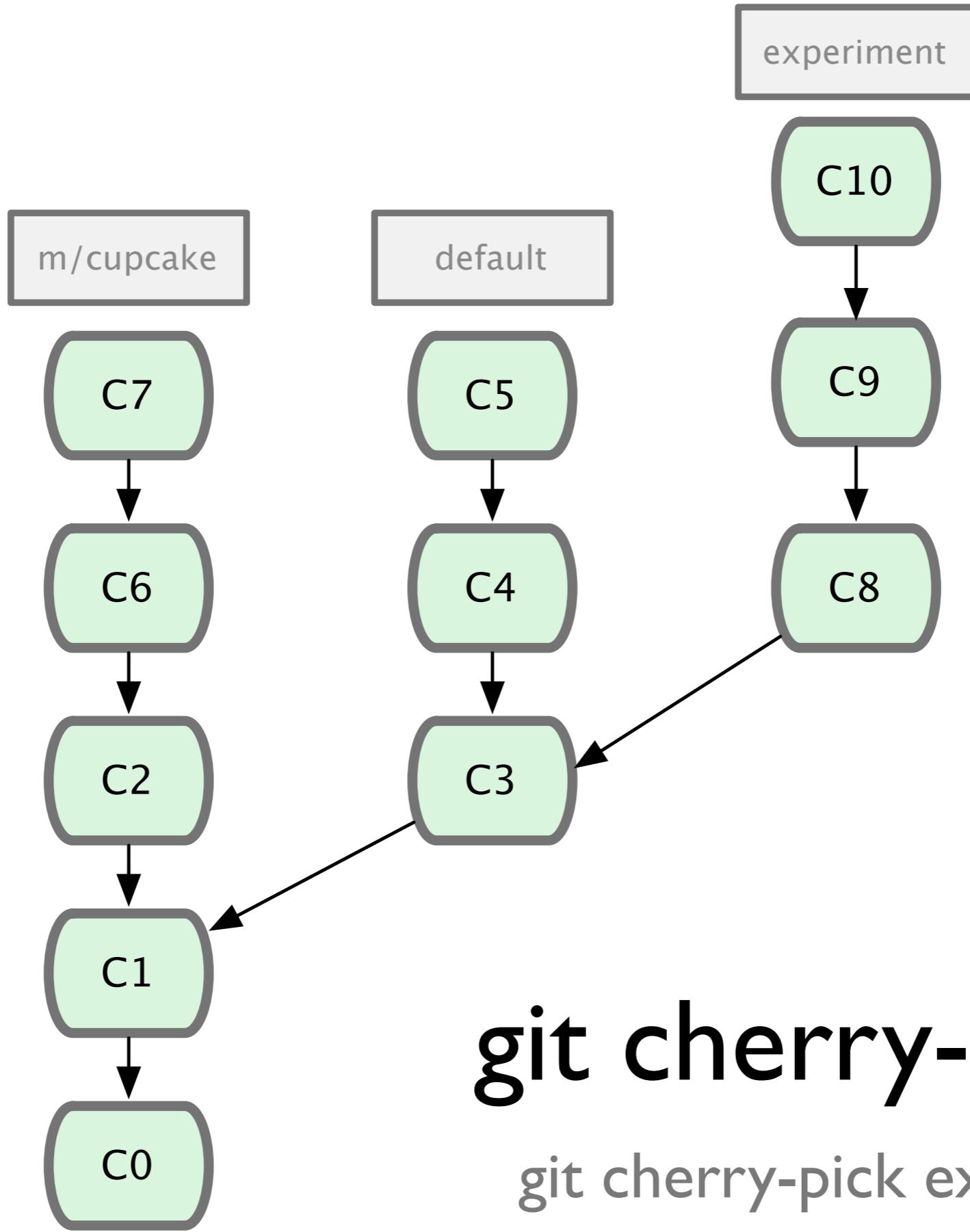




git checkout default

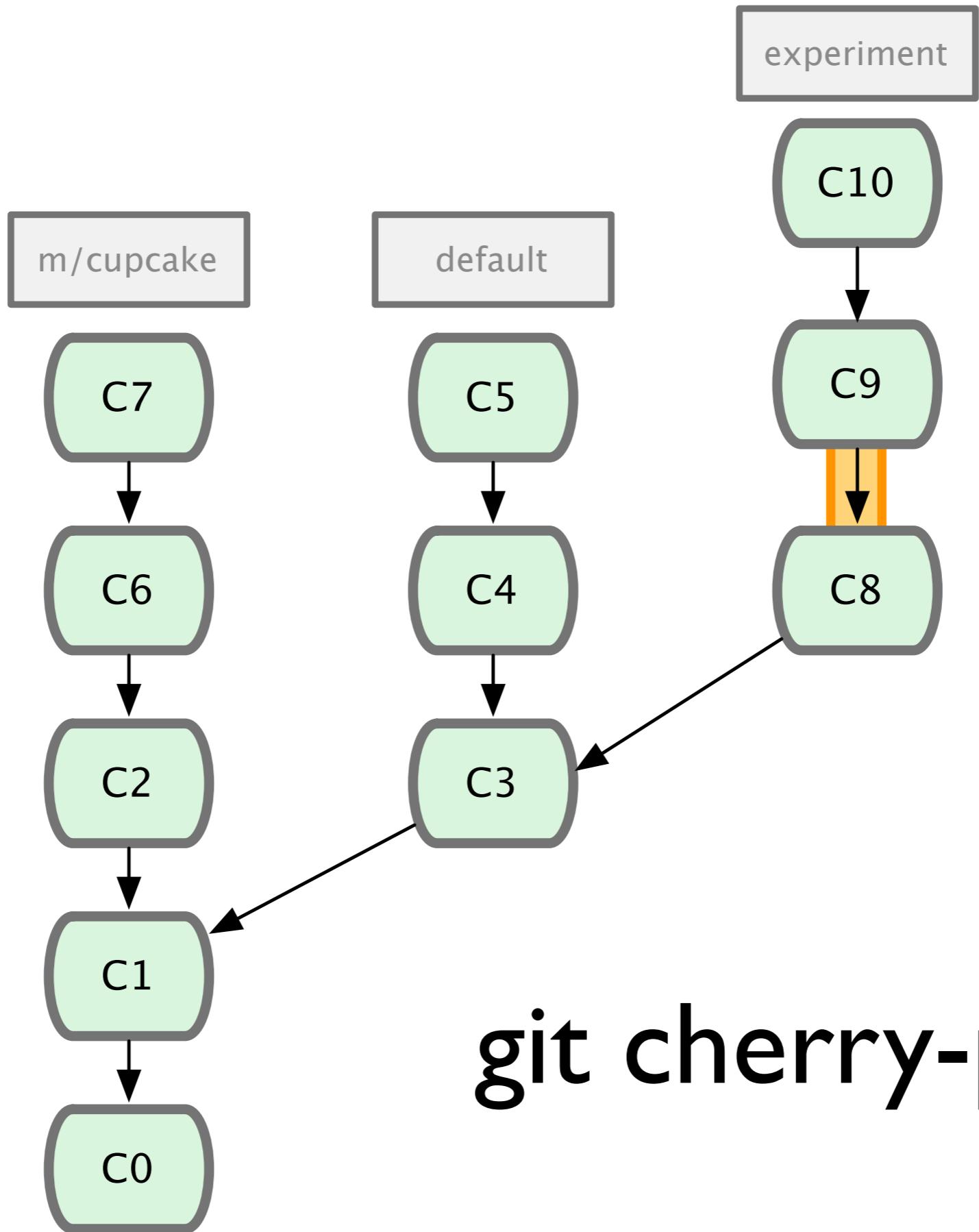


git cherry-pick c9

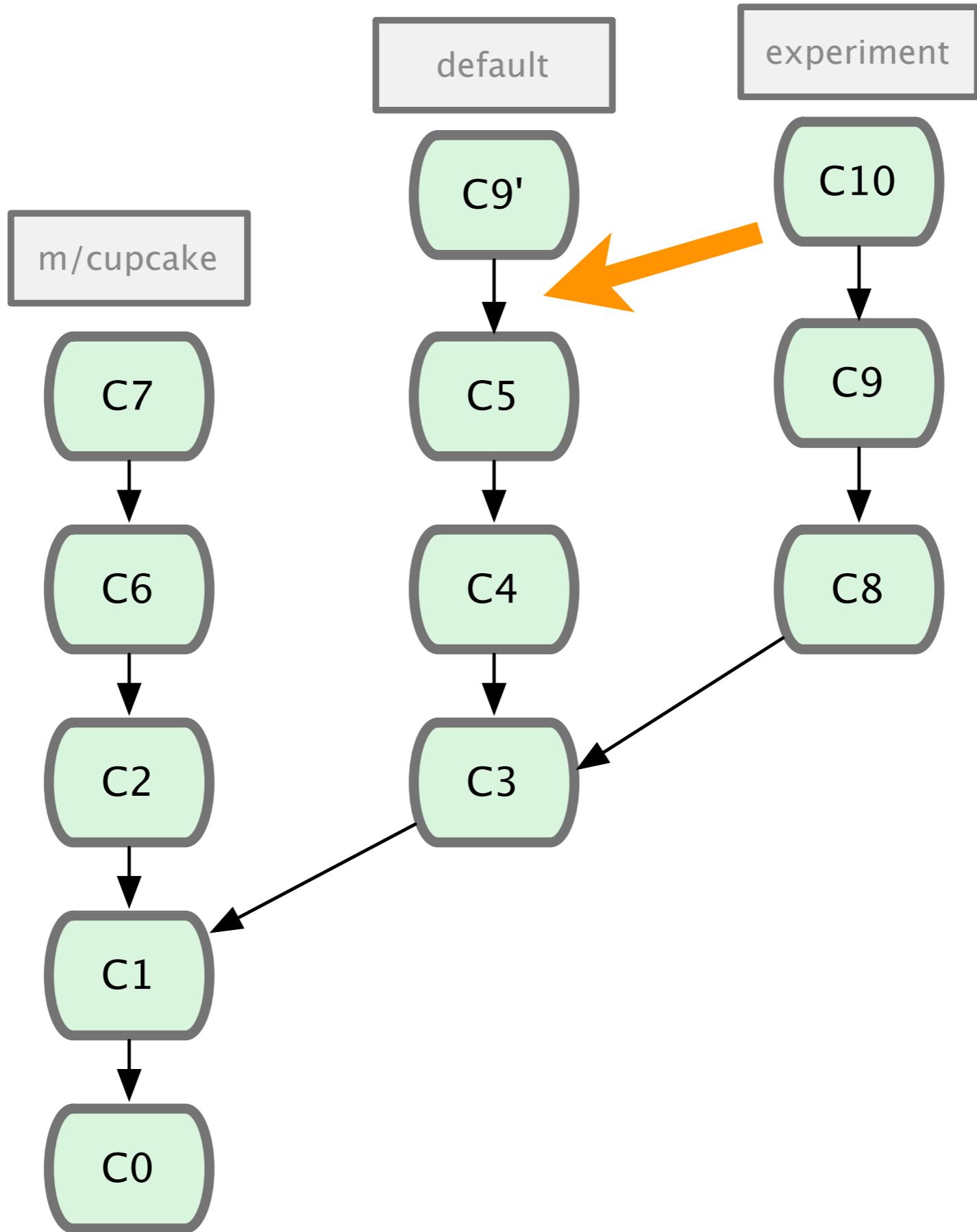


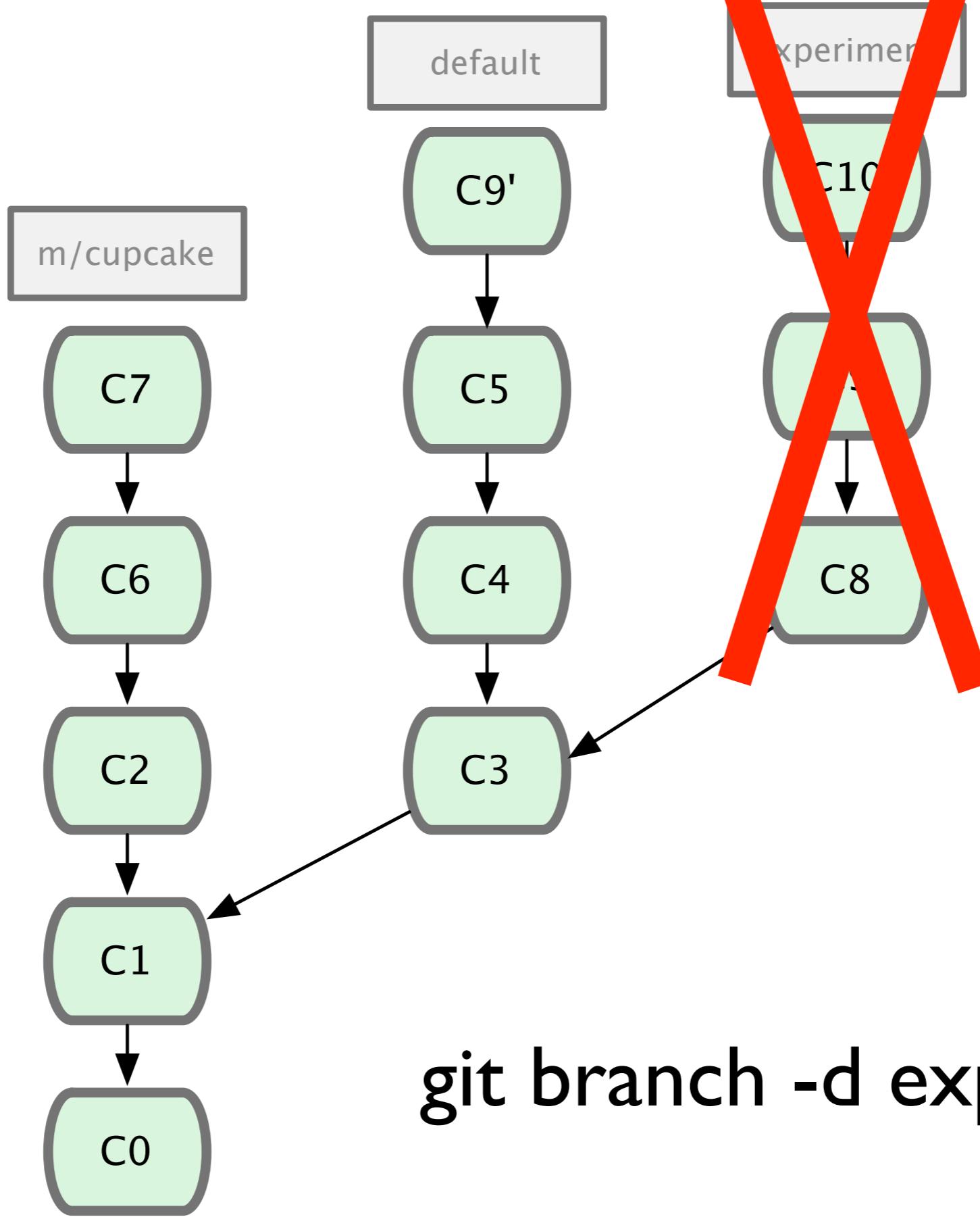
git cherry-pick c9

git cherry-pick experiment^

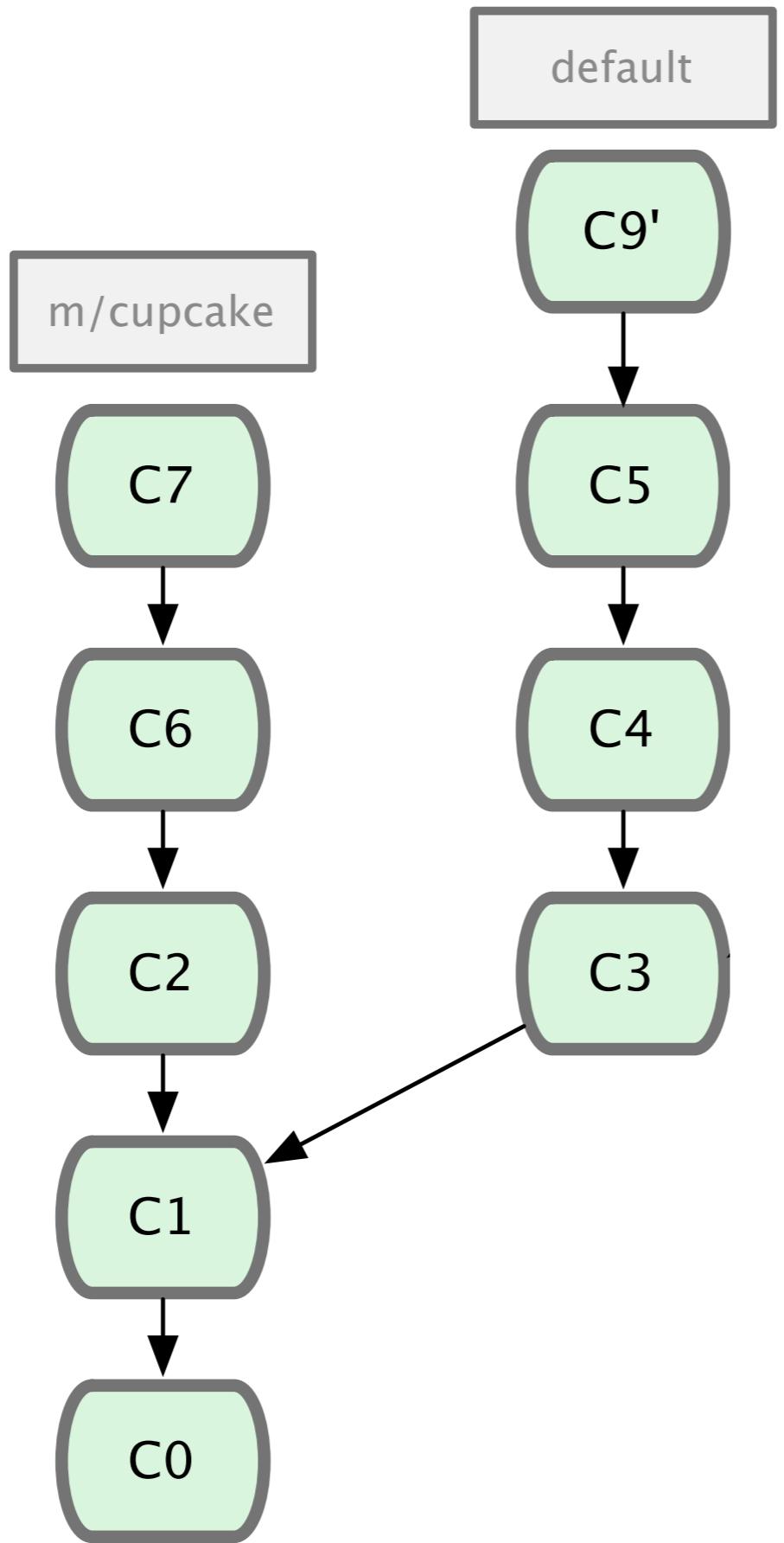


git cherry-pick c9





git branch -d experiment



Customizing

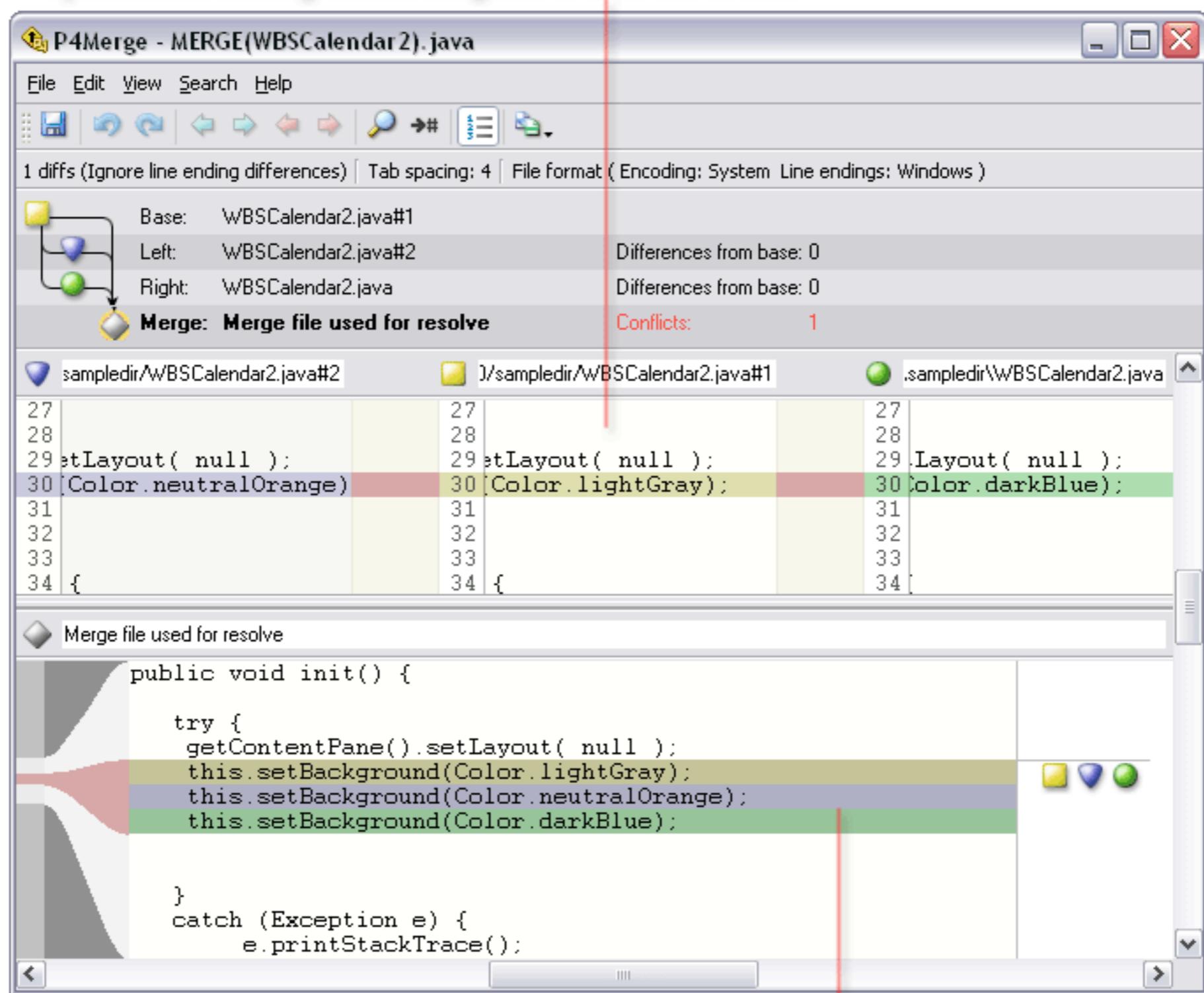
Colors

```
$ git config --global color.ui true
```

Custom Merge Tool

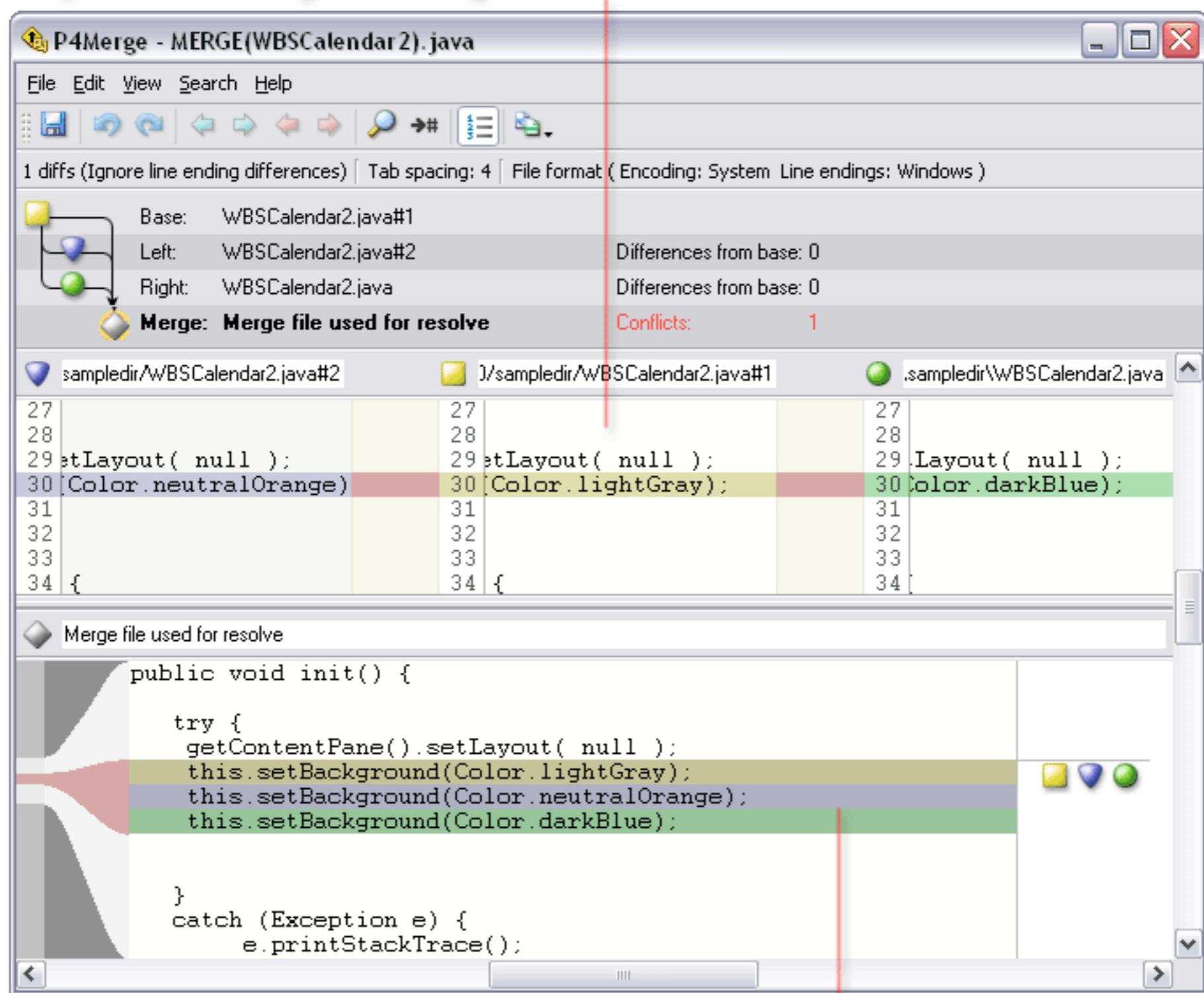
perform visual
merge tool

Synchronized scrolling enables navigation of three files at once



Merged changes can be re-selected and re-edited

Synchronized scrolling enables navigation of three files at once



Merged changes can be re-selected and re-edited

```
$ cat /usr/local/bin/extMerge
#!/bin/sh
/Applications/p4merge.app/Contents/MacOS/p4merge $*
```

```
$ git config --global merge.tool extMerge
```

```
$ git config --global mergetool.extMerge.cmd 'extMerge  
"$BASE" "$LOCAL" "$REMOTE" "$MERGED"'
```

```
$ git config --global mergetool.trustExitCode = false
```

~/.gitconfig

```
[merge]
  tool = extMerge
[mergetool "extMerge"]
  cmd = extMerge "$BASE" "$LOCAL" "$REMOTE" "$MERGED"
  trustExitCode = false
```

git mergetool

Git Attributes

.gitattributes

Diff Binary Files

Images

```
diff --git a/image.png b/image.png
index 88839c4..4afcb7c 100644
Binary files a/image.png and b/image.png differ
```

**tell Git how to diff a
binary file**

exiftool

```
$ echo '*.*.png diff=exif' >> .gitattributes
```

```
$ echo '*.png diff=exif' >> .gitattributes
```

every file that ends in .png

```
$ echo '*.*.png diff=exif' >> .gitattributes
```

pre-process them with a strategy called ‘exif’

```
$ echo '*.png diff=exif' >> .gitattributes  
$ git config diff.exif.textconv exiftool
```

```
$ echo '*.*.png diff=exif' >> .gitattributes  
$ git config diff.exif.textconv exiftool
```

the ‘exif’ strategy is to run ‘exiftool’ on it

```
$ exiftool image.png
ExifTool Version Number      : 7.74
File Name                   : image.png
Directory                   : .
File Size                   : 94 kB
File Modification Date/Time : 2009:04:21 07:02:43-07:00
File Type                   : PNG
MIME Type                   : image/png
Image Width                 : 1056
Image Height                : 827
Bit Depth                   : 8
Color Type                  : RGB with Alpha
Compression                 : Deflate/Inflate
Filter                      : Adaptive
Interlace                    : Noninterlaced
Profile CMM Type           : appl
Profile Version             : 2.0.0
Profile Class               : Display Device Profile
Color Space Data            : RGB
Profile Connection Space    : XYZ
Profile Date Time           : 2009:04:05 12:26:58
Profile File Signature       : acsp
Primary Platform             : Apple Computer Inc.
CMM Flags                   : Not Embedded, Independent
Device Manufacturer          :
Device Model                 :
Device Attributes            : Reflective, Glossy, Positive, Color
Rendering Intent              : Perceptual
Connection Space Illuminant   : 0.9642 1 0.82491
Profile Creator               : appl
Profile ID                   : 0
Red Matrix Column             : 0.39493 0.22505 0.02957
Green Matrix Column            : 0.42793 0.69208 0.14424
Blue Matrix Column             : 0.14134 0.08327 0.65096
Media White Point              : 0.95047 1 1.0891
Chromatic Adaptation          : 1.04788 0.02292 -0.0502 0.02957 0.99049 -0.01706 -0.00923 0.01508 0.75165
Red Tone Reproduction Curve    : (Binary data 14 bytes, use -b option to extract)
Green Tone Reproduction Curve   : (Binary data 14 bytes, use -b option to extract)
Blue Tone Reproduction Curve     : (Binary data 14 bytes, use -b option to extract)
```

```
$ echo '*.png diff=exif' >> .gitattributes  
$ git config diff.exif.textconv exiftool
```

```
diff --git a/image.png b/image.png
index 88839c4..4afcb7c 100644
--- a/image.png
+++ b/image.png
@@ -1,12 +1,12 @@
    ExifTool Version Number          : 7.74
-File Size                         : 70 kB
-File Modification Date/Time      : 2009:04:21 07:02:45-07:00
+File Size                         : 94 kB
+File Modification Date/Time      : 2009:04:21 07:02:43-07:00
    File Type                        : PNG
    MIME Type                        : image/png
-Image Width                       : 1058
-Image Height                      : 889
+Image Width                       : 1056
+Image Height                      : 827
    Bit Depth                        : 8
    Color Type                       : RGB with Alpha
```

Documents

```
$ echo '*.doc diff=doc' >> .gitattributes  
$ git config diff.doc.textconv strings
```

```
$ git diff
diff --git a/chapter1.doc b/chapter1.doc
index c1c8a0a..b93c9e4 100644
--- a/chapter1.doc
+++ b/chapter1.doc
@@ -8,7 +8,8 @@ re going to cover Version Control System
re going to cover how to get it and set it up for the -
t already have it on your system.

In Chapter Two we will go over basic Git usage - how to
-s going on, modify stuff and contribute changes. If the
+s going on, modify stuff and contribute changes. If the
+Let's see if this works.

Chapter Three is about the branching model in Git, often
```

```
$ git diff  
diff --git a/chapter1.doc b/chapter1.doc  
index c1c8a0a..b93c9e4 100644  
--- a/chapter1.doc  
+++ b/chapter1.doc  
@@ -8,7 +8,8 @@ re going to cover Version Control System  
re going to cover how to get it and set it up for the -  
t already have it on your system.  
In Chapter Two we will go over basic Git usage - how to  
-s going on, modify stuff and contribute changes. If the  
+s going on, modify stuff and contribute changes. If the  
+Let's see if this works.
```

Chapter Three is about the branching model in Git, often

File Filtering

Staging Area

fileA.txt

fileB.txt

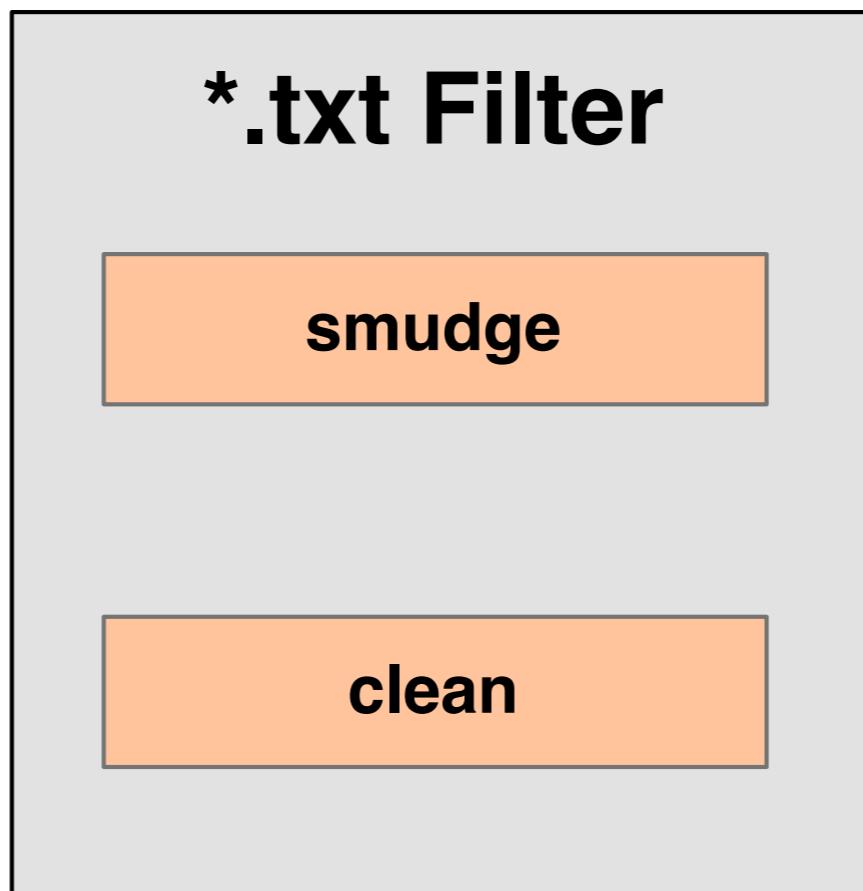
fileC.rb

Staging Area

fileA.txt

fileB.txt

fileC.rb

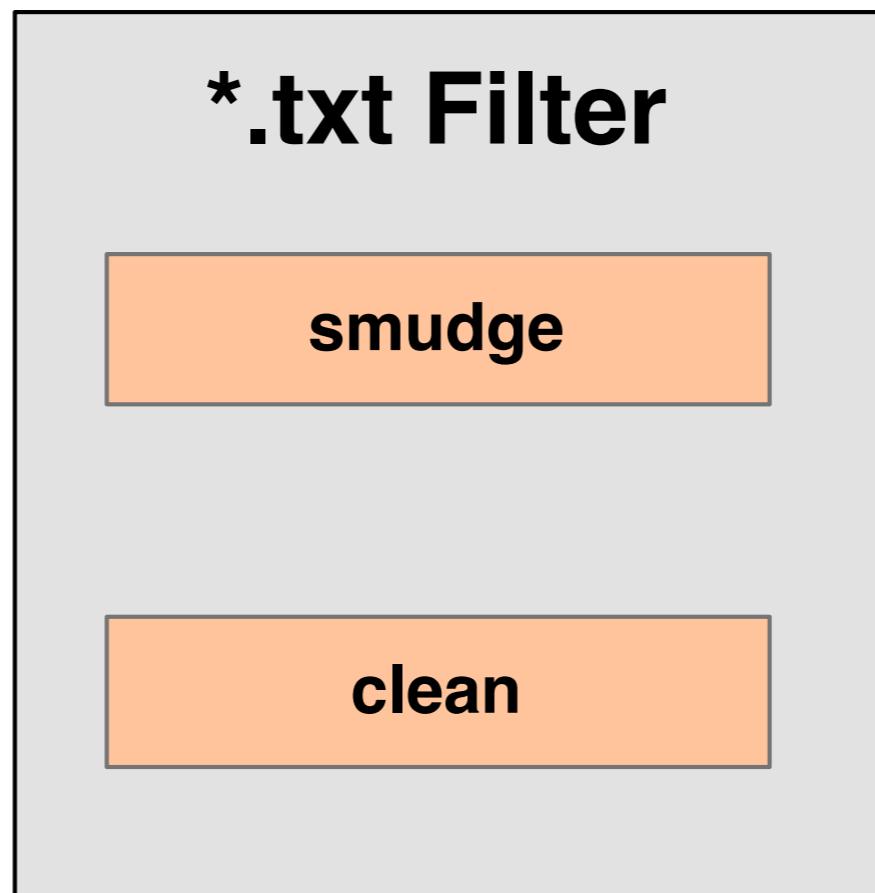


Staging Area

fileA.txt

fileB.txt

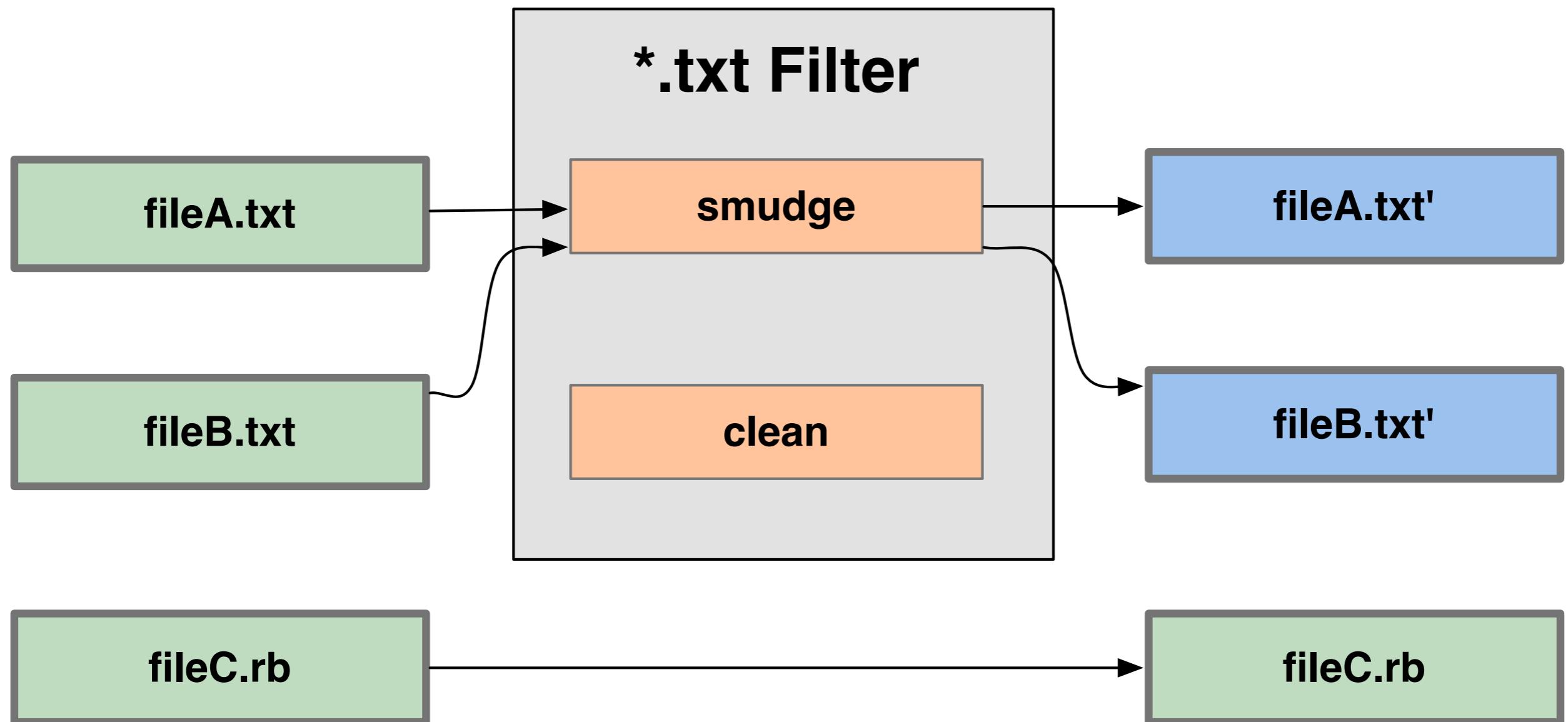
fileC.rb



git checkout

Staging Area

Working Directory



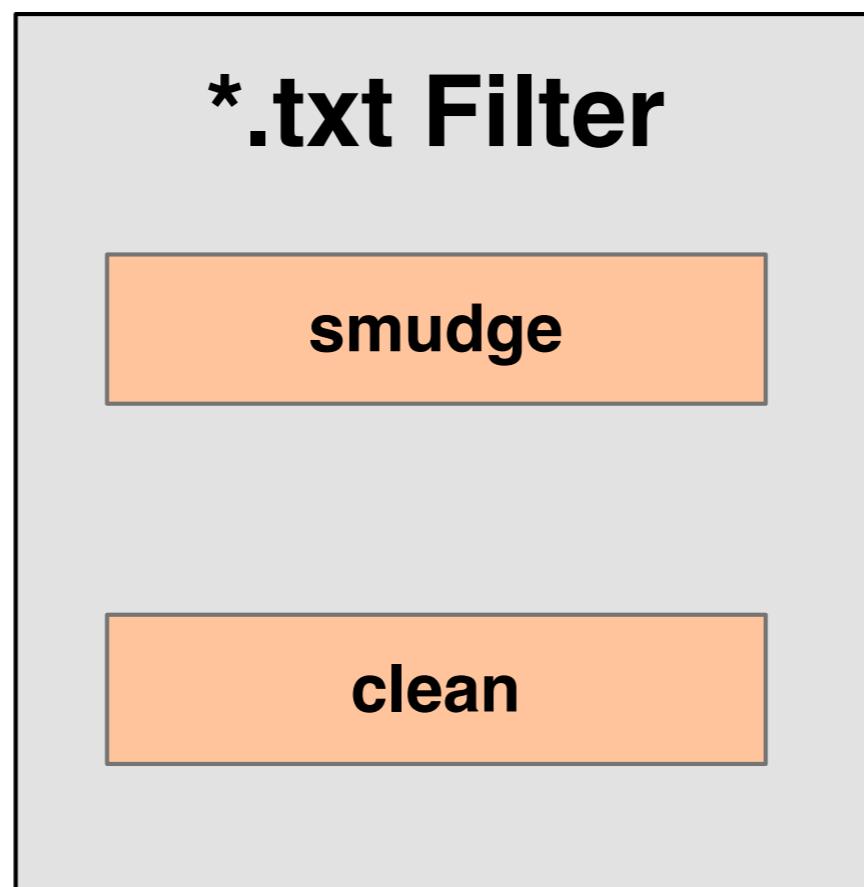
git checkout

Staging Area

fileA.txt

fileB.txt

fileC.rb



Working Directory

fileA.txt'

fileB.txt'

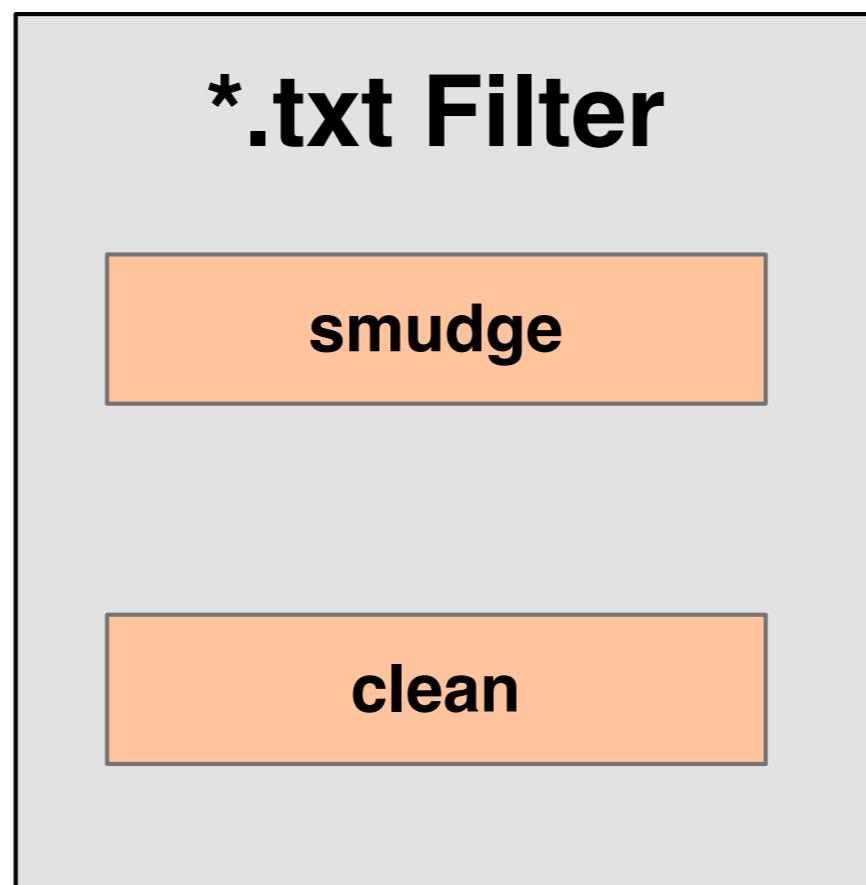
fileC.rb

Staging Area

fileA.txt

fileB.txt

fileC.rb



Working Directory

fileA.txt'

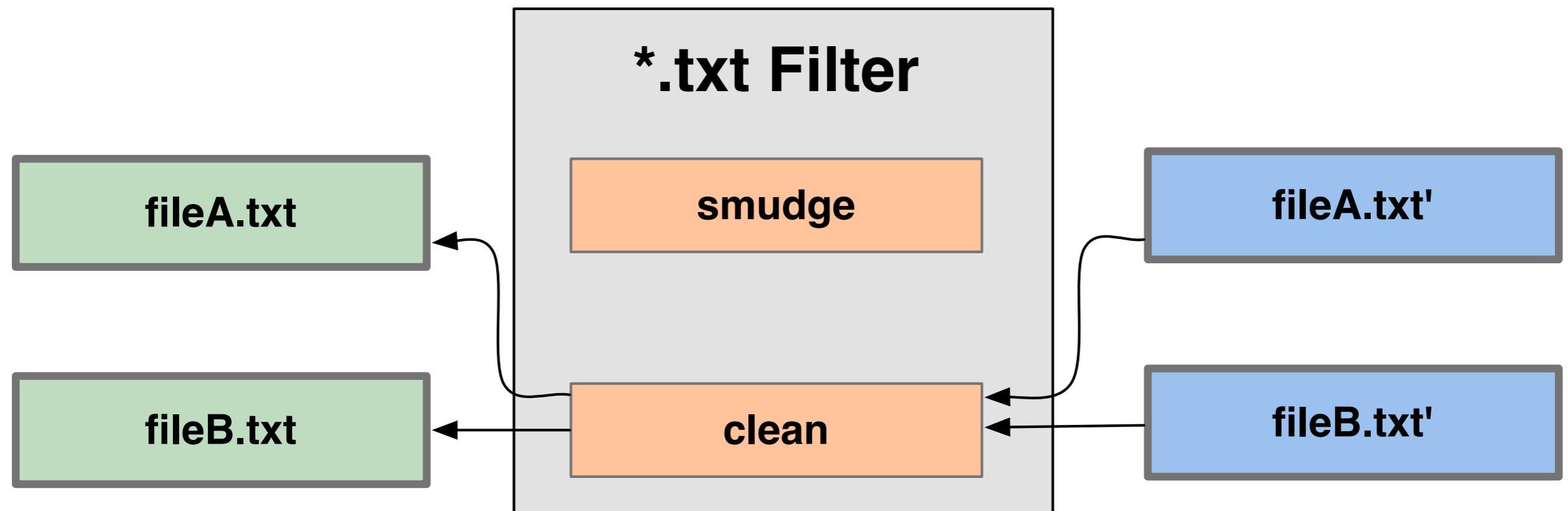
fileB.txt'

fileC.rb

git commit

Staging Area

Working Directory



git commit

expanding a \$Date\$

/usr/bin/expand_date

```
#!/usr/bin/env ruby
data = STDIN.read
date = `git log --pretty=format:"%ad" -1`
puts data.gsub('$Date$', '$Date: ' + date + '$')
```

```
git config filter.dater.smudge expand_date
```

```
git config filter.dater.smudge expand_date
```

```
git config filter.dater.clean
```

```
'perl -pe "s/\\\\$Date[^\\\\$]*\\\$\$/\\\\\$Date\\\\$/"'
```

replace \$Date(whatever)\$ with \$Date\$

```
git config filter.dater.smudge expand_date
```

```
git config filter.dater.clean
```

```
'perl -pe "s/\\\\\$Date[^\\\\\\$]*\\\\$/\\\\\\\$Date\\\\\$/"'
```

replace **\$Date(whatever)\$** with **\$Date\$**

test it

```
$ echo '# $Date$' > date_test.rb
```

```
$ echo '# $Date$' > date_test.rb
$ echo 'date*.rb filter=dater' >> .gitattributes
```

```
$ echo '# $Date$' > date_test.rb  
$ echo 'date*.rb filter=dater' >> .gitattributes
```

use the ‘dater’ filter for any files matching ‘date*.rb’

```
$ echo '# $Date$' > date_test.rb
$ echo 'date*.rb filter=dater' >> .gitattributes

$ git add date_test.rb .gitattributes
$ git commit -m "Testing date expansion in Git"
```

```
$ echo '# $Date$' > date_test.rb
$ echo 'date*.rb filter=dater' >> .gitattributes

$ git add date_test.rb .gitattributes
$ git commit -m "Testing date expansion in Git"
$ rm date_test.rb
$ git checkout date_test.rb
```

```
$ echo '# $Date$' > date_test.rb
$ echo 'date*.rb filter=dater' >> .gitattributes

$ git add date_test.rb .gitattributes
$ git commit -m "Testing date expansion in Git"
$ rm date_test.rb
$ git checkout date_test.rb

$ cat date_test.rb
# $Date: Tue Apr 21 07:26:52 2009 -0700$
```

```
$ echo '# $Date$' > date_test.rb
$ echo 'date*.rb filter=dater' >> .gitattributes

$ git add date_test.rb .gitattributes
$ git commit -m "Testing date expansion in Git"
$ rm date_test.rb
$ git checkout date_test.rb

$ cat date_test.rb
# $Date: Tue Apr 21 07:26:52 2009 -0700$
```