

Getting Git

Scott Chacon

10%
4 min

2m

This Slide
Who is Scott?

75%
35 min

20m

How Does Git Work?

15%
6 min

4m

How Do I Use Git?
How Do I Deploy with Git?
Where Can I Learn More?

2m

10%
4 min

2m

This Slide

75%
35 min

2m

Who is Scott?

15%
6 min

3m

What is Git?

20m

How Does Git Work?

12m

How Do I Use Git?

4m

How Do I Deploy with Git?

2m

Where Can I Learn More?

10%
4 min

2m

This Slide

2m

Who is Scott?

75%
35 min

3m

What is Git?

20m

How Does Git Work?

12m

How Do I Use Git?

15%
6 min

4m

How Do I Deploy with Git?

2m

Where Can I Learn More?

10%
4 min

2m

This Slide
Who is Scott?

75%
35 min

20m

How Does Git Work?

15%
6 min

4m

How Do I Use Git?
How Do I Deploy with Git?
Where Can I Learn More?

2m

10%
4 min

2m

This Slide
Who is Scott?

75%
35 min

20m

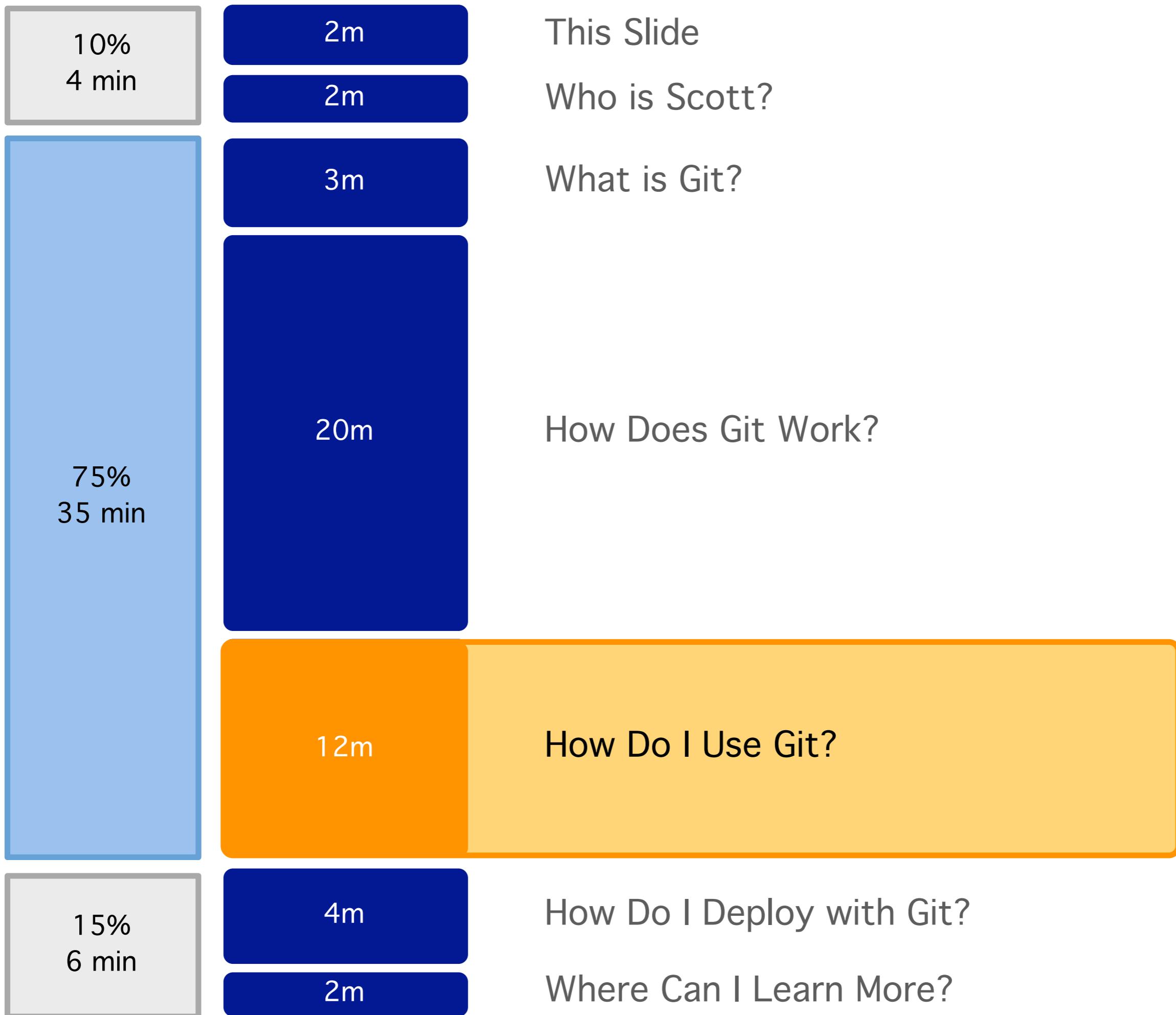
How Does Git Work?

15%
6 min

4m

How Do I Use Git?
How Do I Deploy with Git?
Where Can I Learn More?

2m



10%
4 min

2m

This Slide
Who is Scott?

75%
35 min

20m

How Does Git Work?

15%
6 min

4m

How Do I Deploy with Git?

2m

Where Can I Learn More?

12m

How Do I Use Git?

10%
4 min

2m

This Slide
Who is Scott?

75%
35 min

3m

What is Git?

20m

How Does Git Work?

12m

How Do I Use Git?

15%
6 min

4m

How Do I Deploy with Git?

2m

Where Can I Learn More?

Who Is Scott?

Introducing myself...

10%
4 min

2m

This Slide

2m

Who is Scott?

75%
35 min

20m

How Does Git Work?

15%
6 min

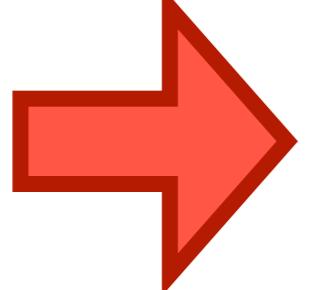
4m

How Do I Use Git?

2m

Where Can I Learn More?

Scott Anthony Chacon



github.com/schacon

The screenshot shows the GitHub profile page for the user 'schacon'. At the top left is a small square profile picture of a man. To its right is the username 'schacon'. Further to the right are links for 'account', 'profile', 'guides', 'log out', and 'repositories: all | search'. Below the profile picture is a blue envelope icon followed by the number '0', indicating zero unread messages.

Your Repositories (create a new one)

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

asgit

fireeaglet

git-lighthouse

git-ruby

git-source

munger

ruby-git

simplegit

ticgit

PeepCode
press

\$9

Git Internals

Source code control and beyond

by Scott Chacon

GitCasts

Browsing Git Objects

May
23

In this episode, I show how to browse and inspect raw Git objects. The major tools covered are the git cat-file and git ls-tree commands to inspect the object contents, and then I cover some of the included graphical browsers, gitk and gitweb.

[watch episode](#) (7:26)

Git Log

May
19

This episode is on git-log, which demonstrates most of the major features and options to the git-log command. It includes showing the stat, short-stat and name-stat options, the —pretty options, the since and until limiters, the path limiter and author field searching.

[watch episode](#) (6:20)

Interactive Adding

May
19

This episode demonstrates how to use the git interactive add command. It covers all of the major features of interactive adding, including status, update, revert, add untracked, patch and diff.



To support GitCasts, and to have a nice searchable PDF of a lot of this stuff, please consider buying my [Git Internals book](#) published by Peepcode Press.



[GitHub](#) : Social Git hosting, a home for your codes.

capistrano

[<-- Thread -->](#)

[<-- Date -->](#)

 Search

[Capis

Jamis Buck

Mon, 12 May 2008 11:02:17 -0700



Welcome new Git Cap Maintainer

Just wanted to drop a note to the list and let everyone know that Scott Chacon, author of the "Git Internals" peepcode PDF (<https://peepcode.com/products/git-internals-pdf>), has volunteered to help maintain capistrano's git SCM module. He'll be acting as a filter for patches to that area of capistrano, making sure the module continues to work in subsequent releases and keeping things sane.

Thanks for volunteering, Scott!

- Jamis

REACTRIX™



Bluge®atibceqeas

jepoirrier@flickr



me

introduce yourself

introduce yourself

and your sweet projects...

why is scott here?

What is Git?



10%
4 min

2m

This Slide
Who is Scott?

75%
35 min

20m

How Does Git Work?

15%
6 min

4m

How Do I Use Git?
How Do I Deploy with Git?
Where Can I Learn More?

2m

git is

**git is
directory content
management system**

git is
tree history
storage system

git is

stupid content tracker

git is

a toolkit

git-add	git-fast-export	git-merge-recur	git-revert
git-add--interactive	git-fast-import	git-merge-recursive	git-rm
git-am	git-fetch	git-merge-recursive-old	git-runstatus
git-annotate	git-fetch--tool	git-merge-resolve	git-send-email
git-apply	git-fetch-pack	git-merge-stupid	git-send-pack
git-applymbox	git-filter-branch	git-merge-subtree	git-sh-setup
git-applypatch	git-fmt-merge-msg	git-merge-tree	git-shell
git-archimport	git-for-each-ref	git-mergetool	git-shortlog
git-archive	git-format-patch	git-mktag	git-show
git-bisect	git-fsck	git-mktree	git-show-branch
git-blame	git-fsck-objects	git-mv	git-show-index
git-branch	git-gc	git-name-rev	git-show-ref
git-bundle	git-get-tar-commit-id	git-pack-objects	git-ssh-fetch
git-cat-file	git-grep	git-pack-redundant	git-ssh-pull
git-check-attr	git-gui	git-pack-refs	git-ssh-push
git-check-ref-format	git-hash-object	git-parse-remote	git-ssh-upload
git-checkout	git-http-fetch	git-patch-id	git-stash
git-checkout-index	git-http-push	git-peek-remote	git-status
git-cherry	git-imap-send	git-prune	git-stripSPACE
git-cherry-pick	git-index-pack	git-prune-packed	git-submodule
git-citool	git-init	git-pull	git-svn
git-clean	git-init-db	git-push	git-svnimport
git-clone	git-instaweb	git-quiltimport	git-symbolic-ref
git-commit	git-local-fetch	git-read-tree	git-tag
git-commit-tree	git-log	git-rebase	git-tar-tree
git-config	git-lost-found	git-rebase--interactive	git-unpack-file
git-convert-objects	git-ls-files	git-receive-pack	git-unpack-objects
git-count-objects	git-ls-remote	git-reflog	git-update-index
git-cvsexportcommit	git-ls-tree	git-relink	git-update-ref
git-cvsimport	git-mailinfo	git-remote	git-update-server-inf
git-cvsserver	git-mailsplit	git-repack	git-upload-archive
git-daemon	git-merge	git-repo-config	git-upload-pack
git-describe	git-merge-base	git-request-pull	git-var
git-diff	git-merge-file	git-rerere	git-verify-pack
git-diff-files	git-merge-index	git-reset	git-verify-tag
git-diff-index	git-merge-octopus	git-resolve	git-web--browse
git-diff-stages	git-merge-one-file	git-rev-list	git-whatchanged
git-diff-tree	git-merge-ours	git-rev-parse	git-write-tree

git-applymbox	git-fast-export git-fast-import	git-merge-recur git-merge-recursive git-merge-recursive-old git-runstatus
git-applypatch	git-fetch--tool git-fetch-pack	git-merge-resolve git-merge-stupid
git-archimport	git-filter-branch git-fmt-merge-msg git-for-each-ref	git-merge-subtree git-merge-tree git-mergetool
	git-fsck git-fsck-objects	git-mktag git-mktree
git-bundle	git-get-tar-commit-id	git-name-rev
git-cat-file		git-pack-objects
git-check-attr		git-pack-redundant
git-check-ref-format	git-hash-object	git-pack-refs
	git-http-fetch	git-parse-remote
git-checkout-index	git-http-push	git-patch-id
git-cherry	git-imap-send git-index-pack	git-peek-remote git-prune git-prune-packed
	git-init-db	
	git-local-fetch	git-quiltimport git-read-tree
git-commit-tree		
git-config	git-lost-found	git-receive-pack
git-convert-objects	git-ls-files	git-reflog
git-count-objects	git-ls-remote	git-relink
git-cvsexportcommit	git-ls-tree	git-remote
git-cvsimport	git-mailinfo	git-repack
git-cvsserver	git-mailsplit	git-repo-config
		git-request-pull
git-describe	git-merge-base	git-rerere
	git-merge-file	
git-diff-files	git-merge-index	git-resolve
git-diff-index	git-merge-octopus	git-rev-list
git-diff-stages	git-merge-one-file	git-rev-parse
git-diff-tree	git-merge-ours	
		git-show-branch git-show-index git-show-ref git-ssh-fetch git-ssh-pull git-ssh-push git-ssh-upload git-stripSPACE git-svn git-svnimport git-symbolic-ref git-tar-tree git-unpack-file git-unpack-objects git-update-index git-update-ref git-update-server-inf git-upload-archive git-upload-pack git-var git-verify-pack git-verify-tag git-web--browse git-whatchanged git-write-tree

git-applybox
git-applypatch
git-archimport

git-bundle
git-cat-file
git-check-attr
git-check-ref-format

git-checkout-index
git-cherry

git-commit-tree
git-config
git-convert-objects
git-count-objects
git-cvsexportcommit
git-cvsimport
git-cvsserver

git-describe

git-diff-files
git-diff-index
git-diff-stages
git-diff-tree

git-fast-export
git-fast-import

git-fetch--tool
git-fetch-pack
git-filter-branch
git-fmt-merge-msg
git-for-each-ref

git-fsck
git-fsck-objects

git-get-tar-commit-id

git-init-db

git-local-fetch

git-lost-found
git-ls-files
git-ls-remote
git-ls-tree
git-mailinfo
git-mailsplit

git-merge-base
git-merge-file
git-merge-index
git-merge-octopus
git-merge-one-file
git-merge-ours

git-merge-recur
git-merge-recursive
git-merge-recursive-old
git-merge-resolve
git-merge-stupid
git-merge-subtree
git-merge-tree
git-mergetool
git-mktag
git-mktree

git-name-rev
git-pack-objects
git-pack-redundant
git-pack-refs
git-patch-object
git-patch-id
git-peer-remote
git-prune
git-prune-packed

git-quiltimport
git-read-tree

git-receive-pack
git-reflog
git-relink
git-remote
git-repack
git-repo-config
git-request-pull
git-rerere

git-resolve
git-rev-list
git-rev-parse

git-runstatus
git-send-pack
git-sh-setup
git-shell
git-shortlog

git-show-branch
git-show-index
git-show-ref
git-ssh-fetch
git-ssh-pull
git-ssh-push
git-ssh-upload

git-strip-space

git-svn
git-svnimport
git-symbolic-ref

git-tar-tree
git-unpack-file
git-unpack-objects
git-update-index
git-update-ref
git-update-server-info
git-upload-archive
git-upload-pack
git-var
git-verify-pack
git-verify-tag
git-web--browse
git-whatchanged
git-write-tree

the “plumbing”

git-add	git-fast-export	git-merge-recur	git-revert
git-add--interactive	git-fast-import	git-merge-recursive	git-rm
git-am	git-fetch	git-merge-recursive-old	git-runstatus
git-annotate	git-fetch--tool	git-merge-resolve	git-send-email
git-apply	git-fetch-pack	git-merge-stupid	git-send-pack
git-applymbox	git-filter-branch	git-merge-subtree	git-sh-setup
git-applypatch	git-fmt-merge-msg	git-merge-tree	git-shell
git-archimport	git-for-each-ref	git-mergetool	git-shortlog
git-archive	git-format-patch	git-mktag	git-show
git-bisect	git-fsck	git-mktree	git-show-branch
git-blame	git-fsck-objects	git-mv	git-show-index
git-branch	git-gc	git-name-rev	git-show-ref
git-bundle	git-get-tar-commit-id	git-pack-objects	git-ssh-fetch
git-cat-file	git-grep	git-pack-redundant	git-ssh-pull
git-check-attr	git-gui	git-pack-refs	git-ssh-push
git-check-ref-format	git-hash-object	git-parse-remote	git-ssh-upload
git-checkout	git-http-fetch	git-patch-id	git-stash
git-checkout-index	git-http-push	git-peek-remote	git-status
git-cherry	git-imap-send	git-prune	git-stripSPACE
git-cherry-pick	git-index-pack	git-prune-packed	git-submodule
git-citool	git-init	git-pull	git-svn
git-clean	git-init-db	git-push	git-svnimport
git-clone	git-instaweb	git-quiltimport	git-symbolic-ref
git-commit	git-local-fetch	git-read-tree	git-tag
git-commit-tree	git-log	git-rebase	git-tar-tree
git-config	git-lost-found	git-rebase--interactive	git-unpack-file
git-convert-objects	git-ls-files	git-receive-pack	git-unpack-objects
git-count-objects	git-ls-remote	git-reflog	git-update-index
git-cvsexportcommit	git-ls-tree	git-relink	git-update-ref
git-cvsimport	git-mailinfo	git-remote	git-update-server-inf
git-cvsserver	git-mailsplit	git-repack	git-upload-archive
git-daemon	git-merge	git-repo-config	git-upload-pack
git-describe	git-merge-base	git-request-pull	git-var
git-diff	git-merge-file	git-rerere	git-verify-pack
git-diff-files	git-merge-index	git-reset	git-verify-tag
git-diff-index	git-merge-octopus	git-resolve	git-web--browse
git-diff-stages	git-merge-one-file	git-rev-list	git-whatchanged
git-diff-tree	git-merge-ours	git-rev-parse	git-write-tree

git-add		git-revert
git-add--interactive		git-rm
git-am	git-fetch	
git-annotate		git-send-email
git-apply		
git-archive	git-format-patch	git-shortlog
git-bisect		git-show
git-blame		
git-branch	git-gc	git-mv
	git-grep	
	git-gui	
git-checkout		git-stash
		git-status
git-cherry-pick		git-submodule
git-citool	git-init	
git-clean		git-pull
git-clone	git-instaweb	git-push
git-commit		
	git-log	git-tag
		git-rebase
		git-rebase--interactive
git-daemon	git-merge	
git-diff		git-reset

git-add
git-add--interactive
git-am
git-annotate
git-apply

git-archive
git-bisect
git-blame
git-branch

git-checkout

git-cherry-pick
git-citool
git-clean
git-clone
git-commit

git-daemon
git-diff

git-fetch
git-format-patch
git-gc
git-grep
git-gui

git-init
git-instaweb
git-log

git-merge

git-revert
git-rm
git-send-email

git-mv

git-submodule

git-stash
git-status

git-tag
git-rebase
git-rebase--interactive

git-reset

the “porcelain”

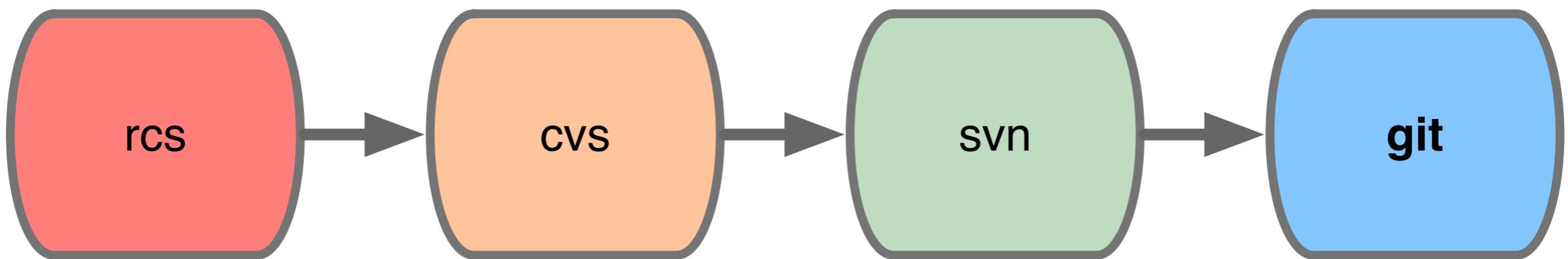
git-add		git-revert
git-add--interactive		git-rm
git-am	git-fetch	
git-annotate		git-send-email
git-apply		
git-archive	git-format-patch	git-shortlog
git-bisect		git-show
git-blame		
git-branch	git-gc	git-mv
	git-grep	
	git-gui	
git-checkout		git-stash
		git-status
git-cherry-pick		git-submodule
git-citool	git-init	
git-clean		git-pull
git-clone	git-instaweb	git-push
git-commit		
	git-log	git-tag
		git-rebase
		git-rebase--interactive
git-daemon	git-merge	
git-diff		git-reset

git is
not subversion!

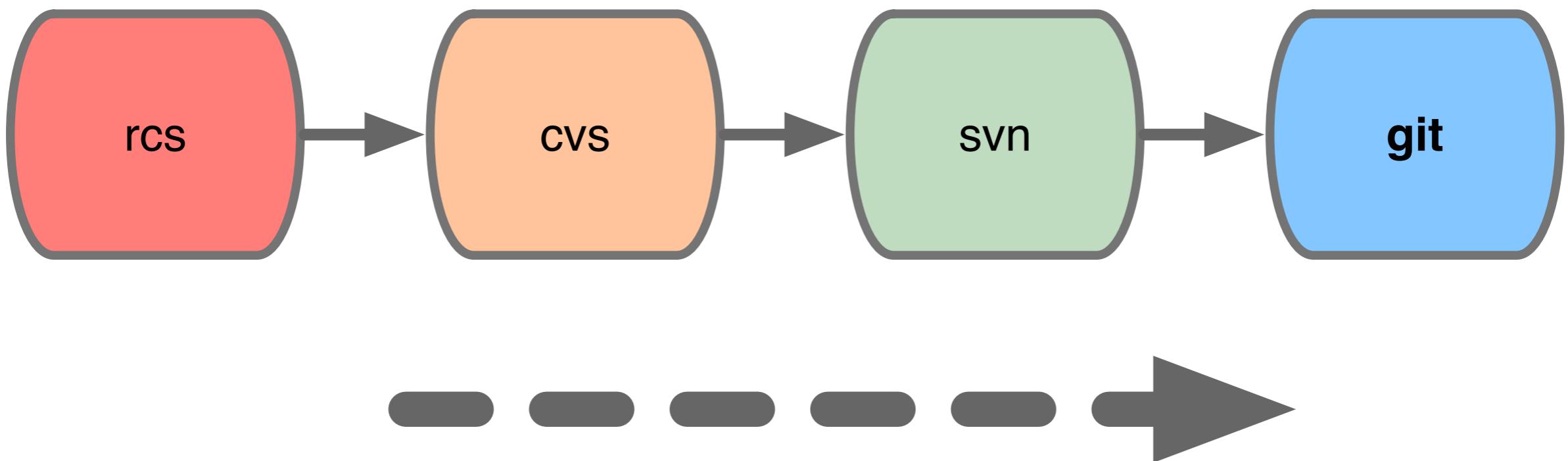
forget subversion!

not an evolution

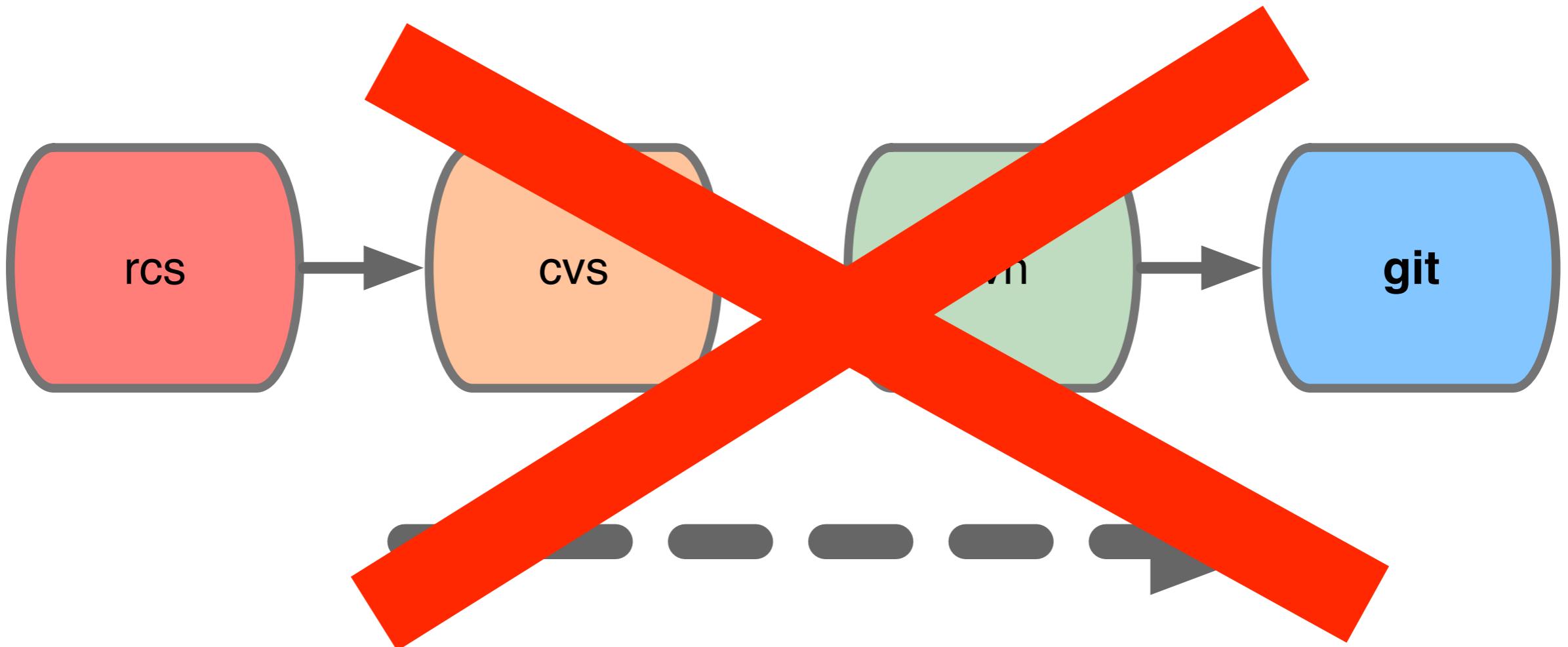
not an evolution



not an evolution



not an evolution



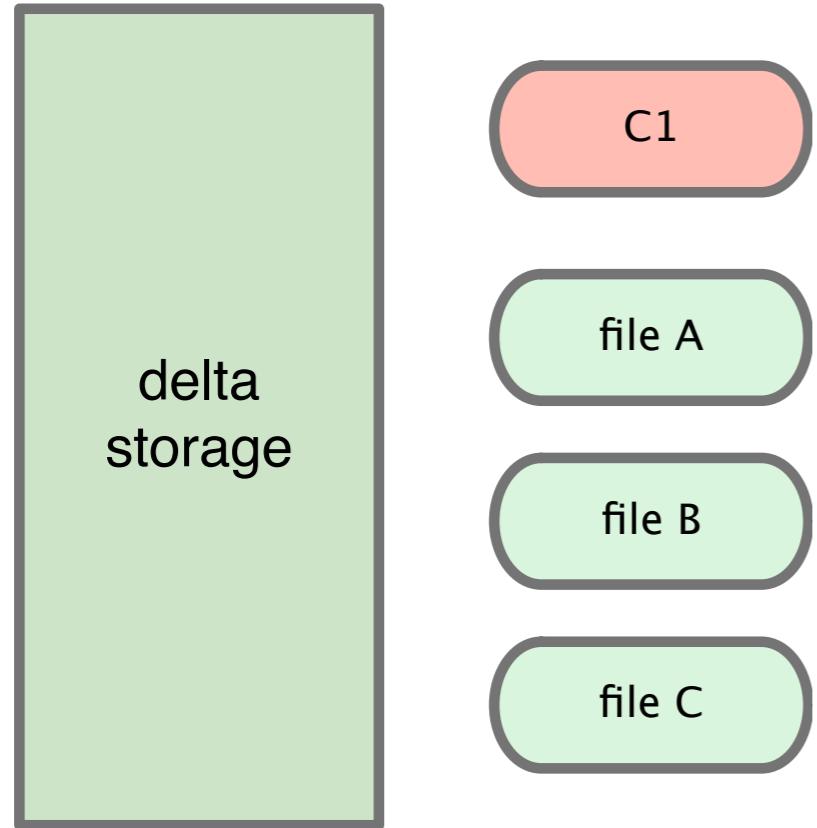
source control taxonomy

source control taxonomy

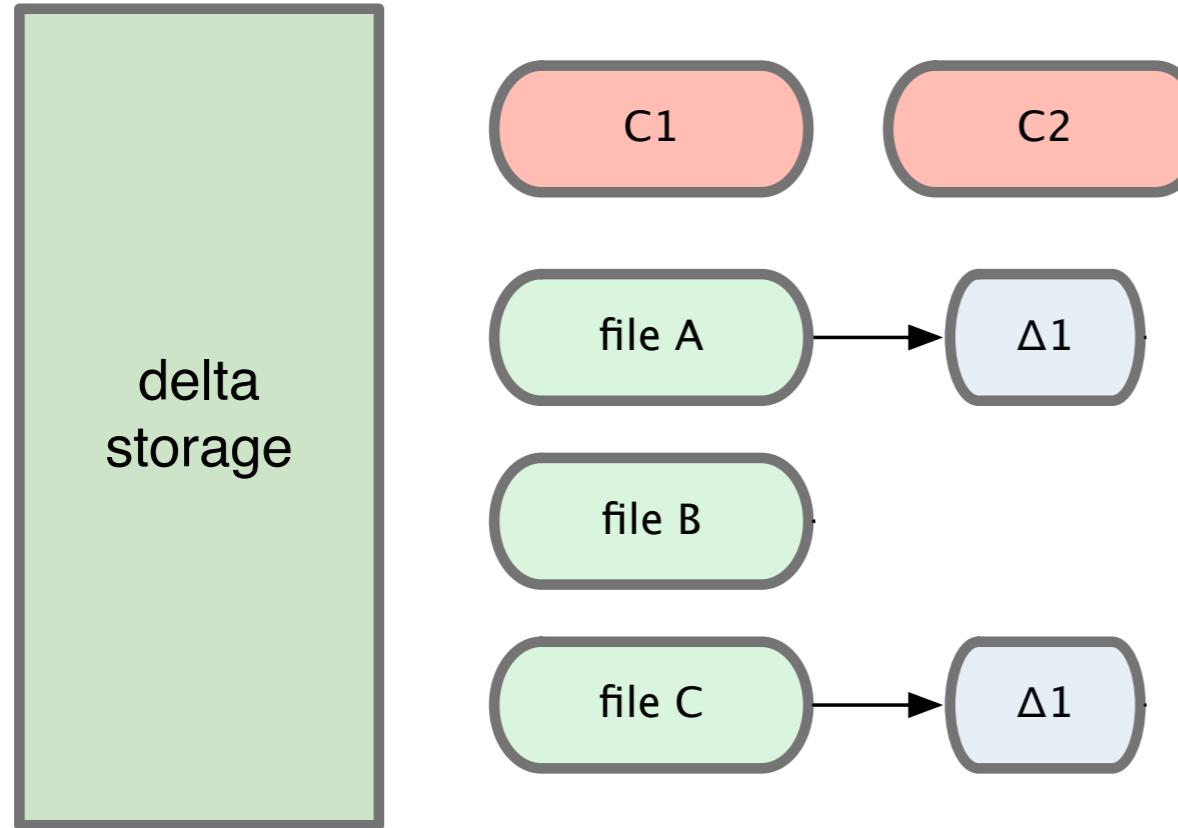


delta
storage

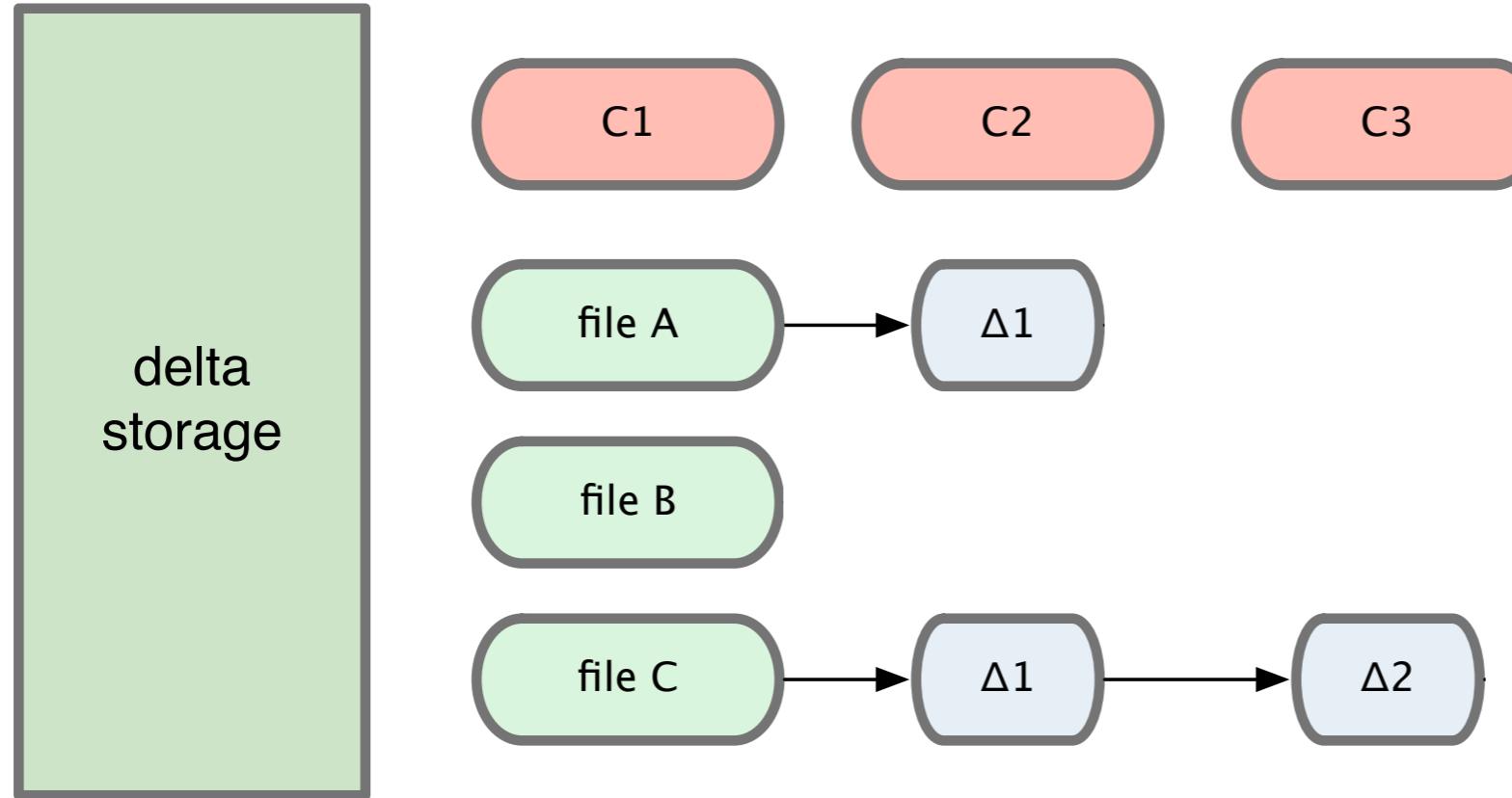
source control taxonomy



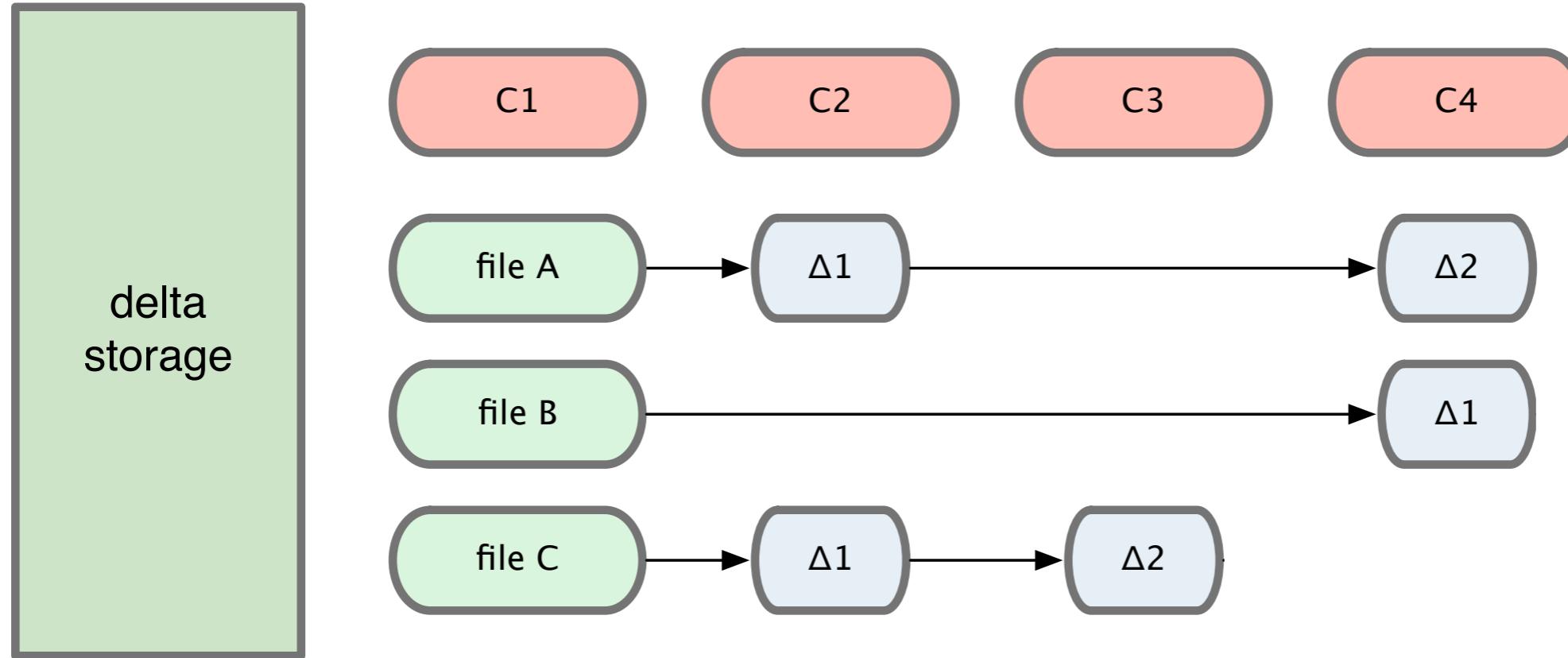
source control taxonomy



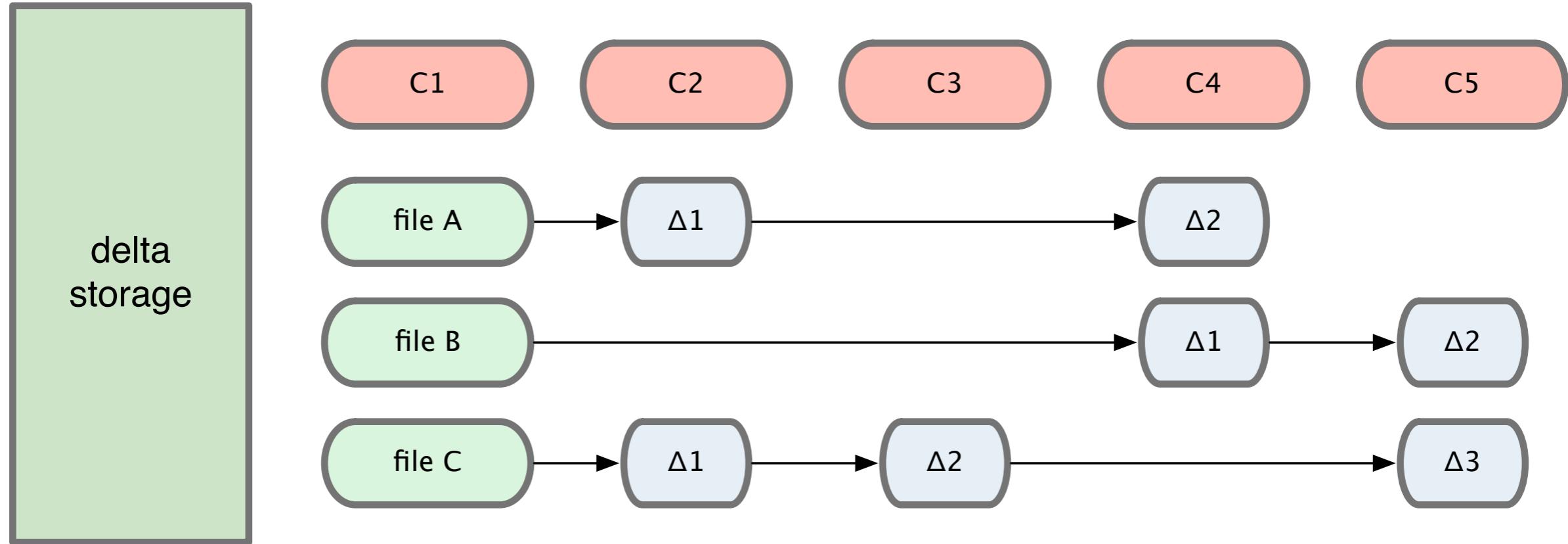
source control taxonomy



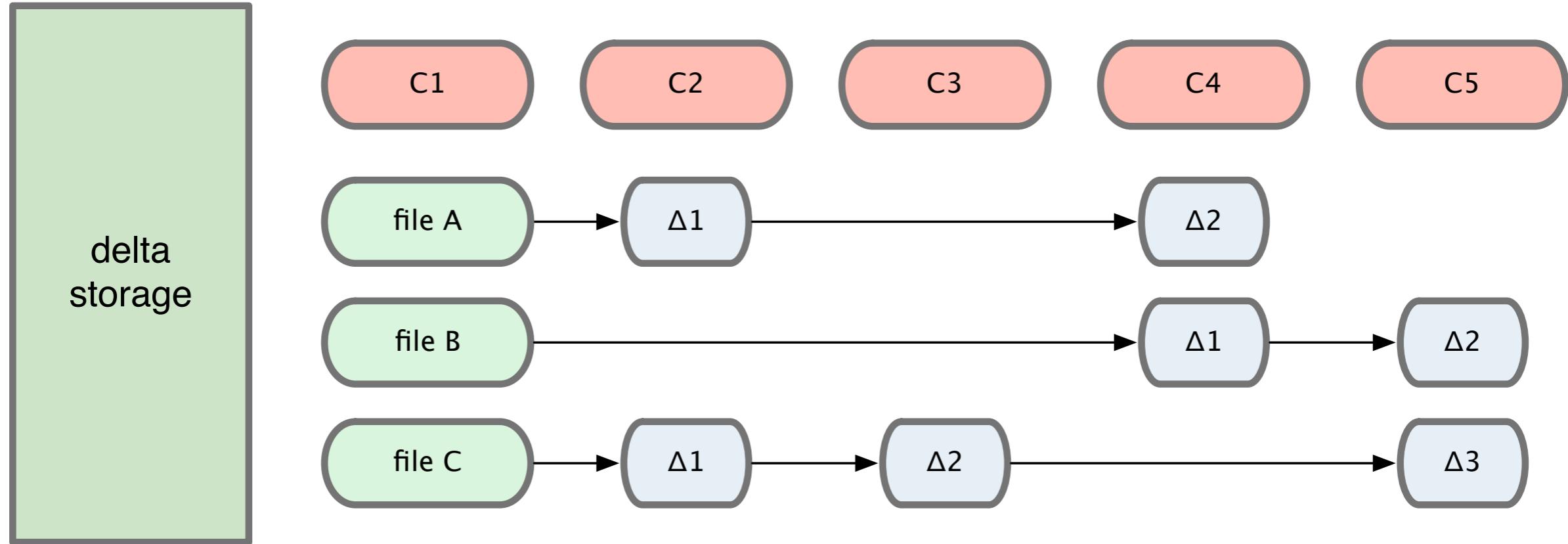
source control taxonomy



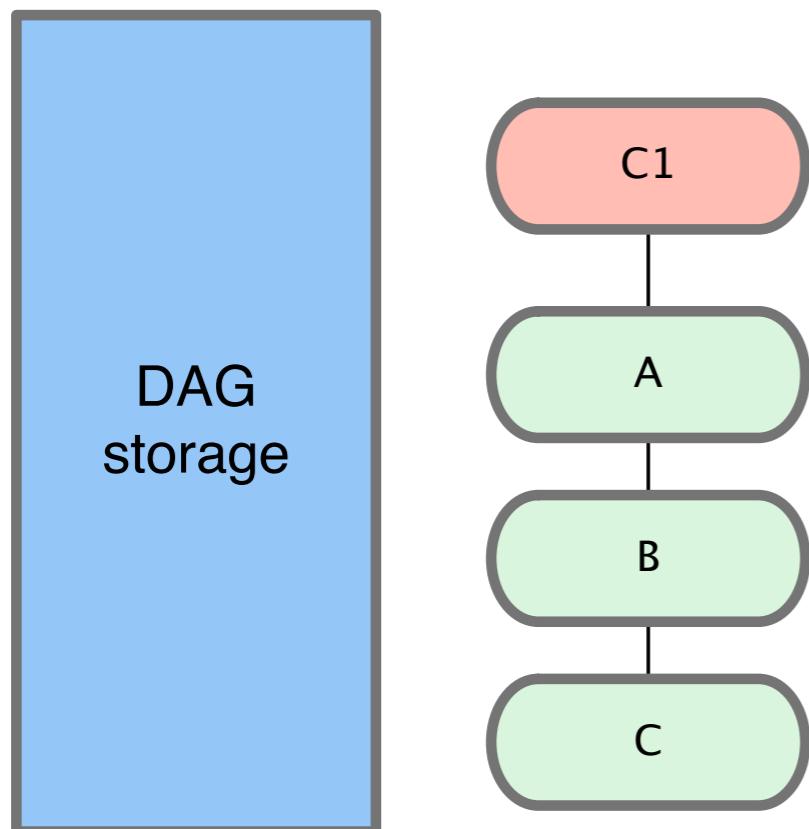
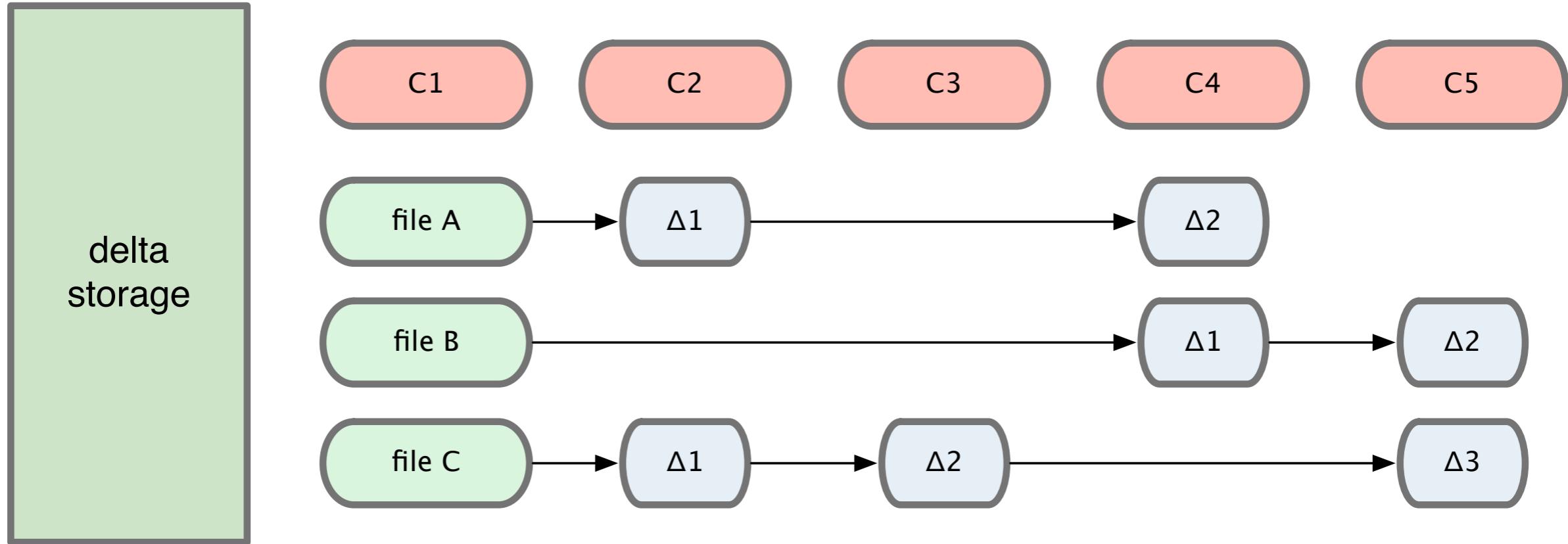
source control taxonomy



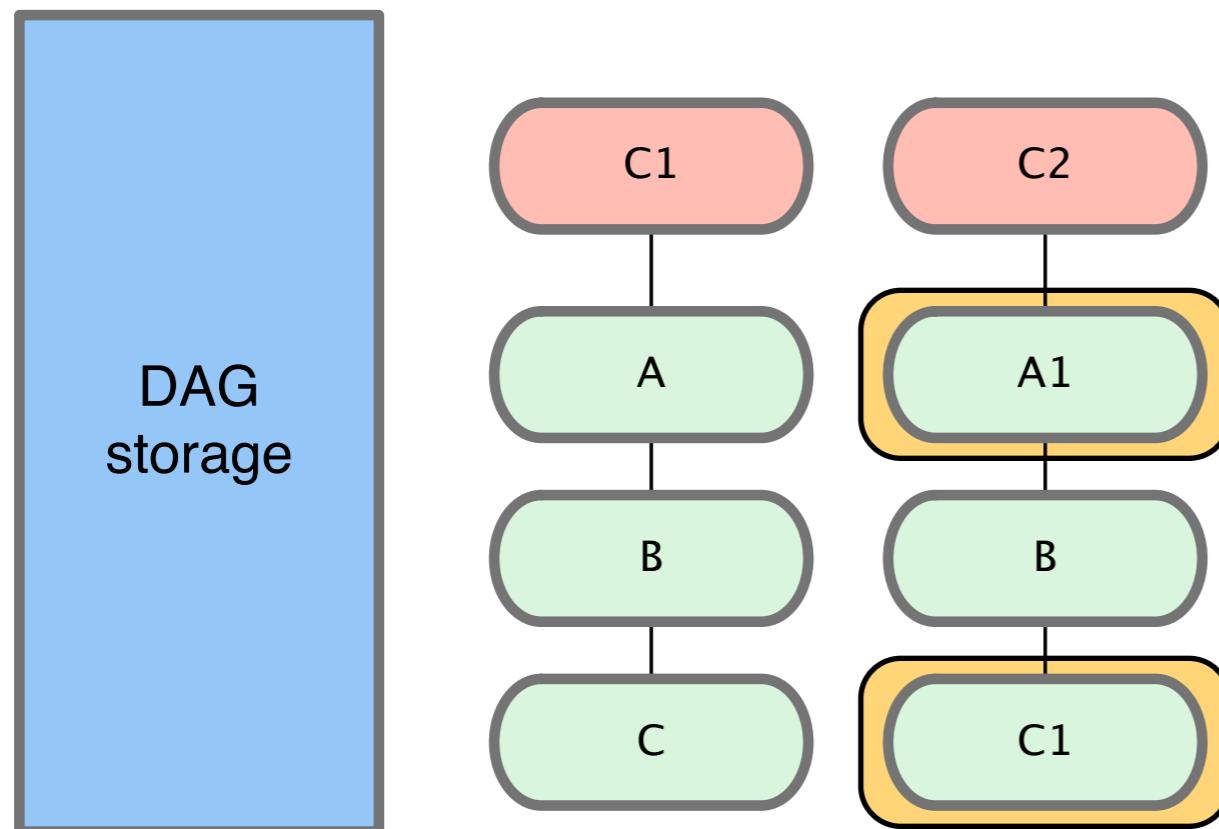
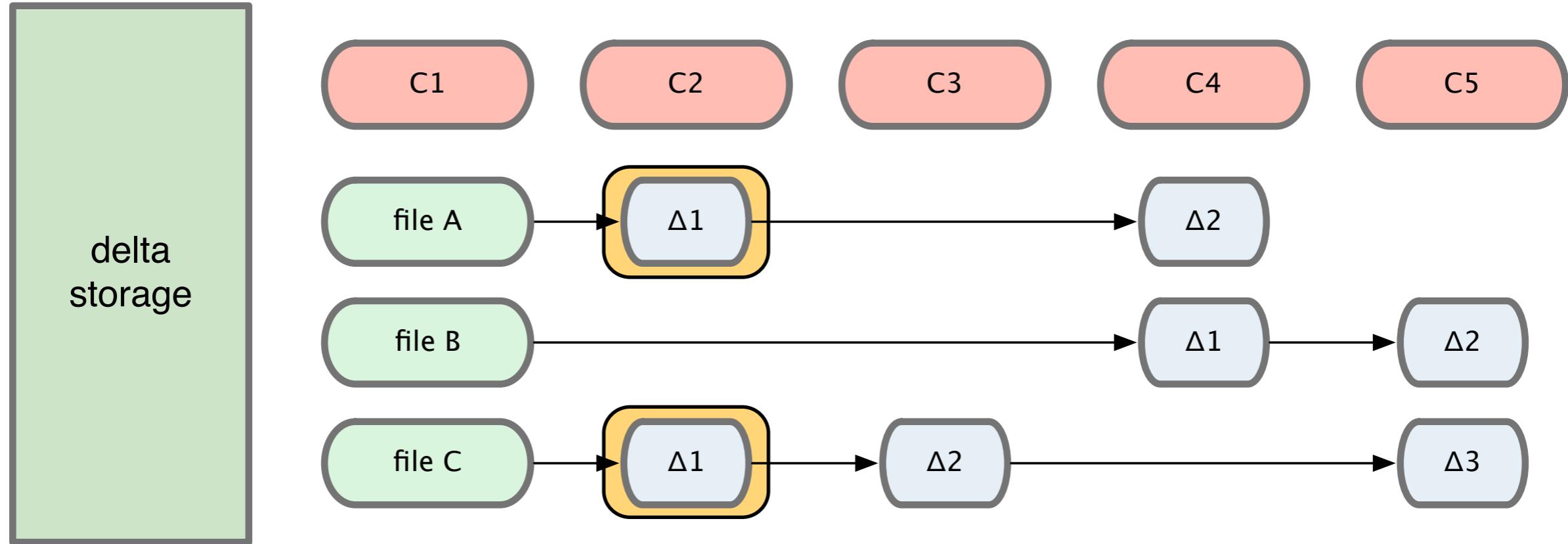
source control taxonomy



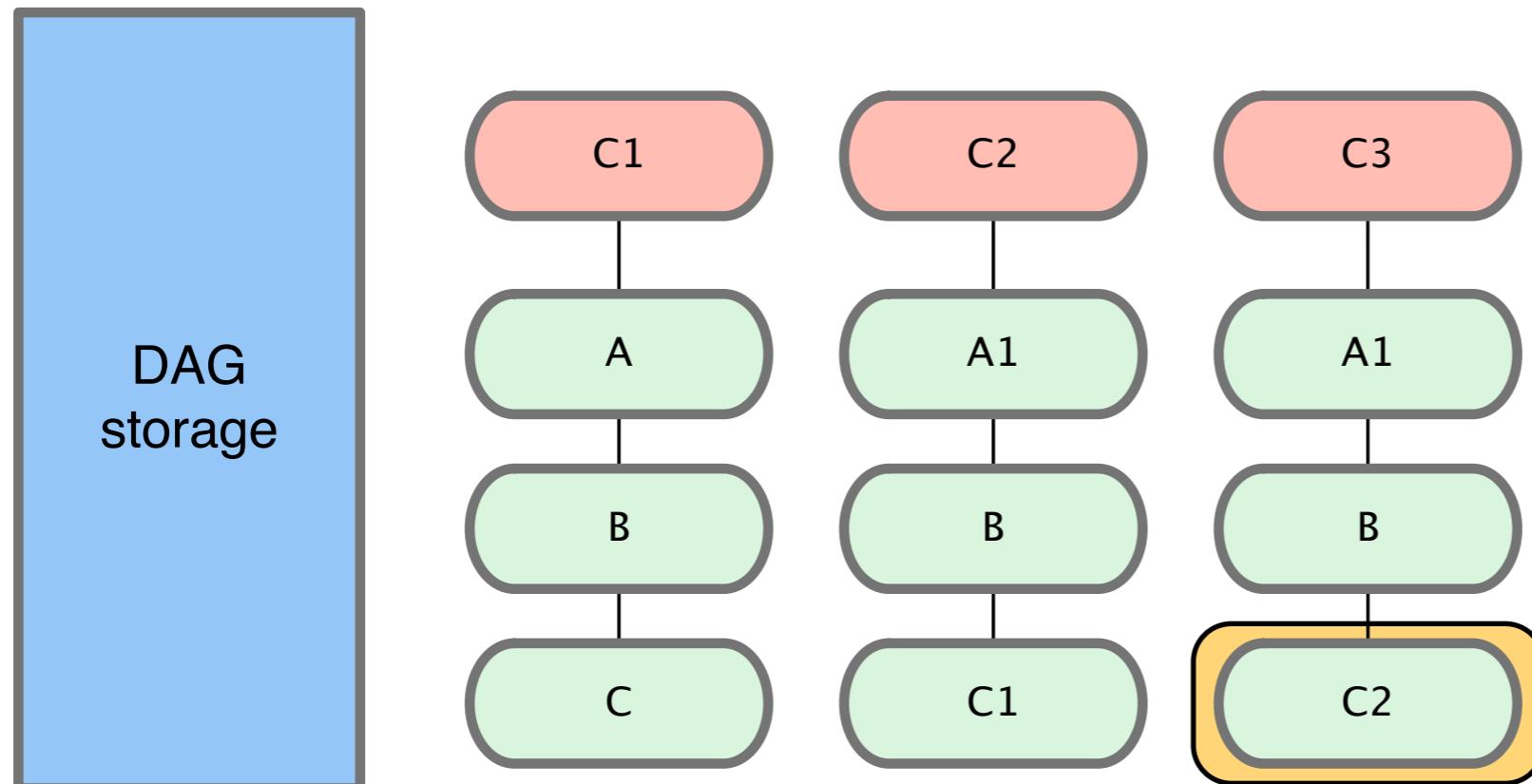
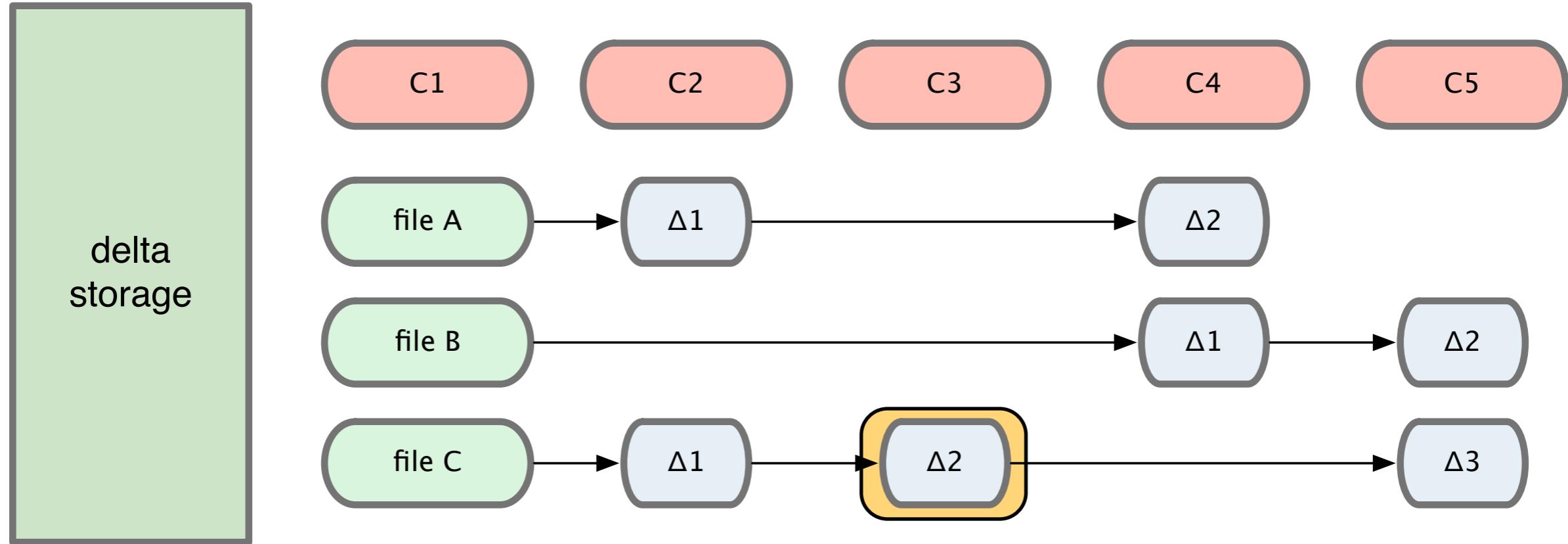
source control taxonomy



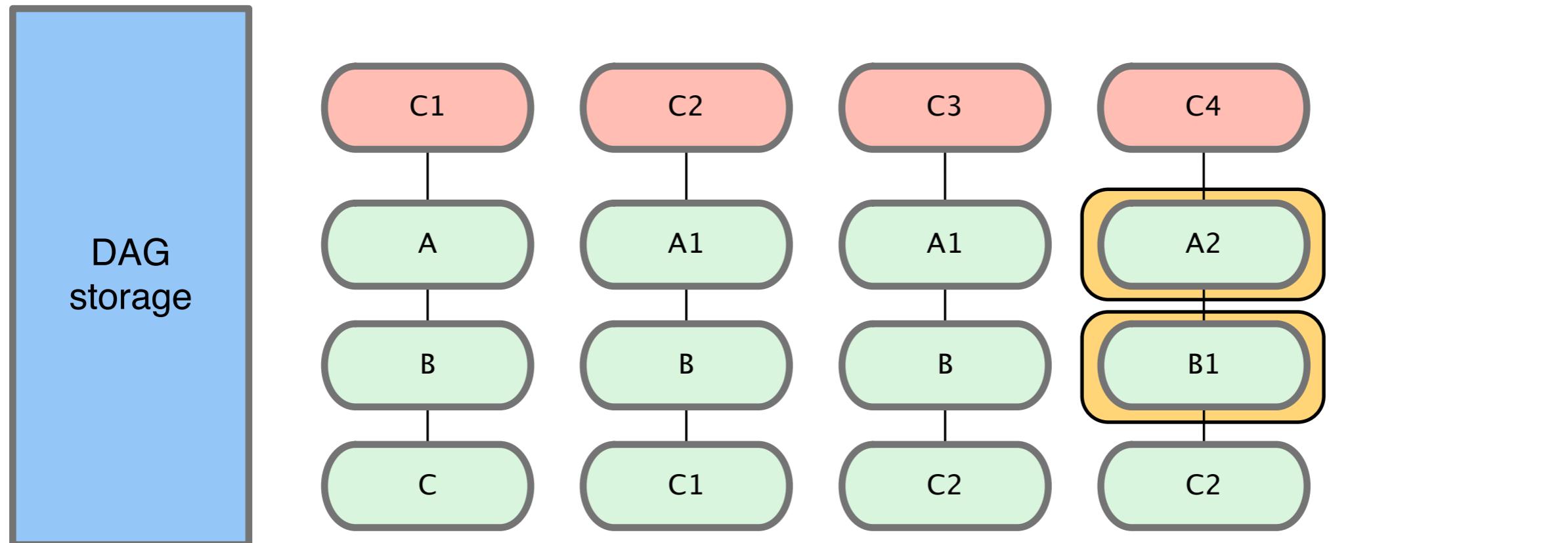
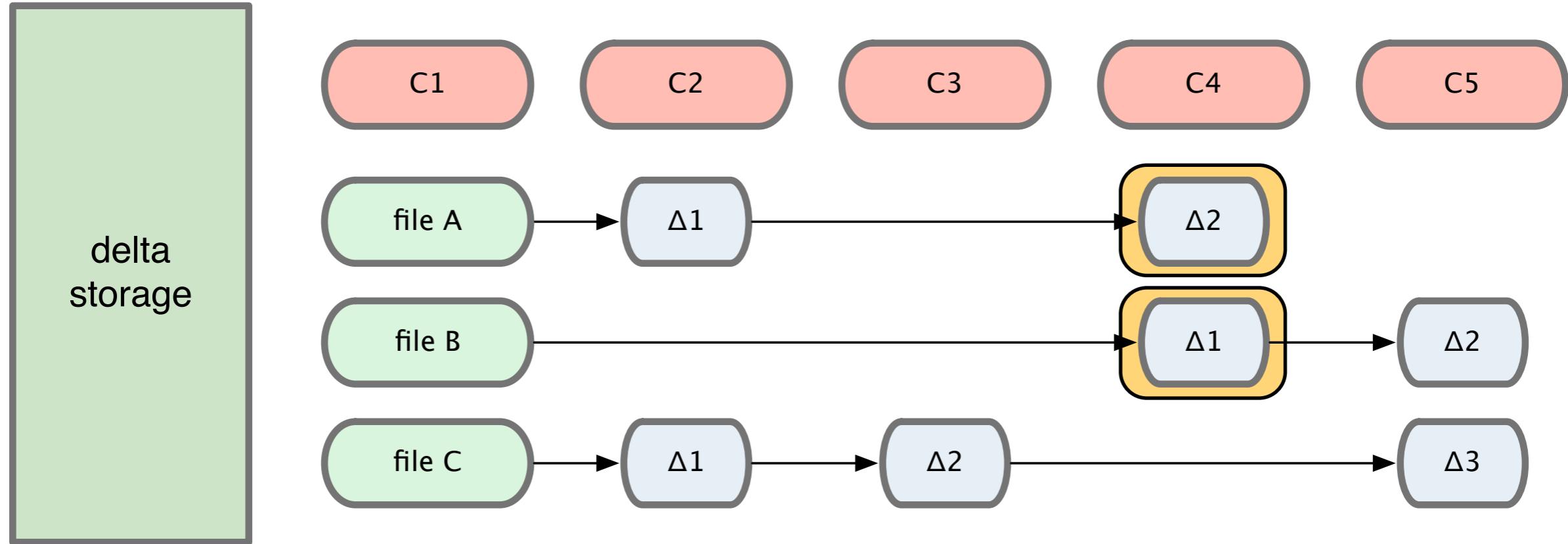
source control taxonomy



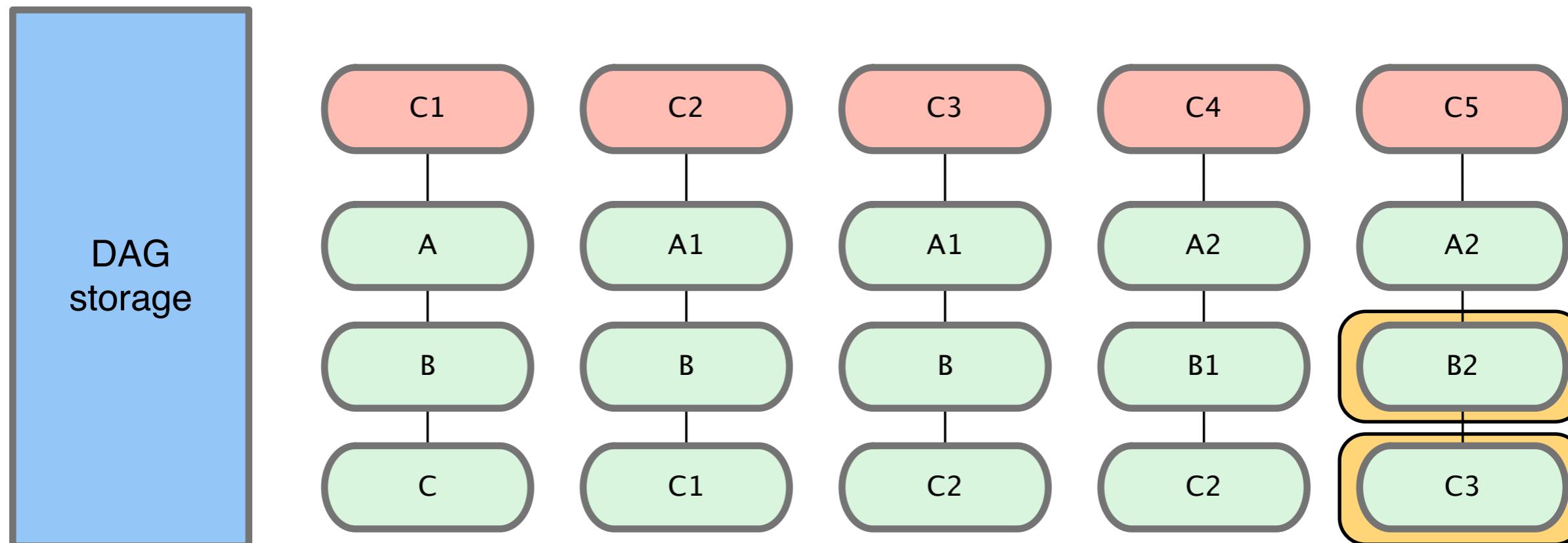
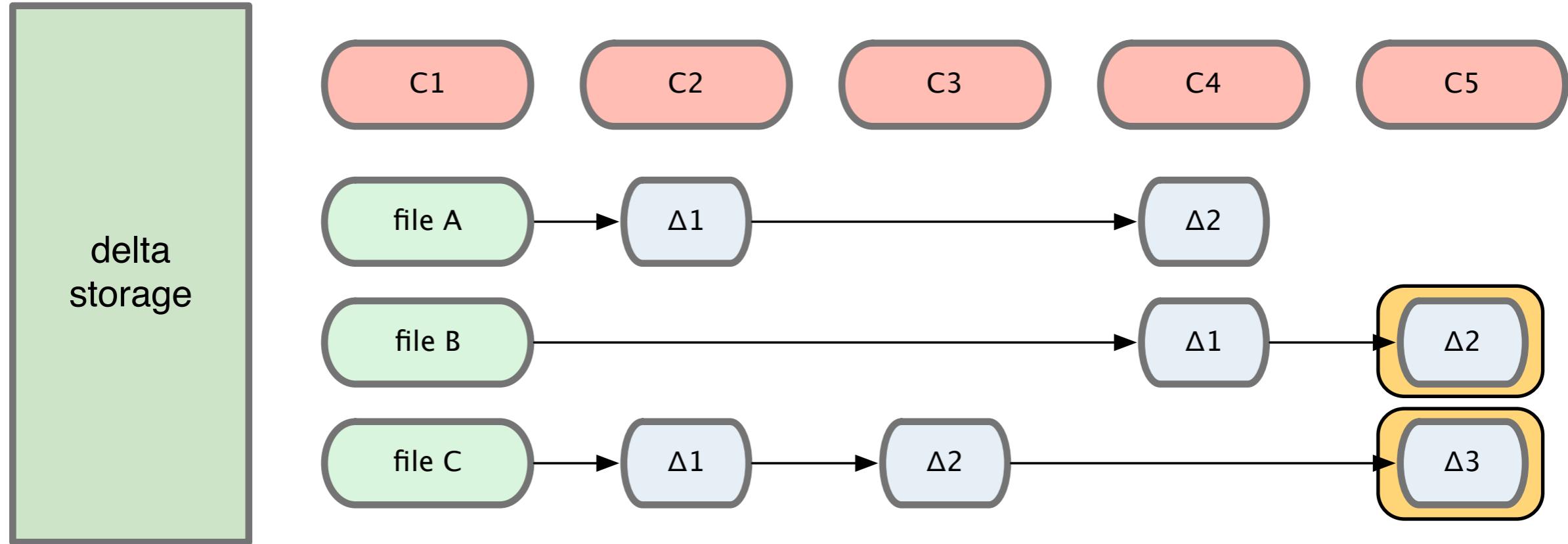
source control taxonomy



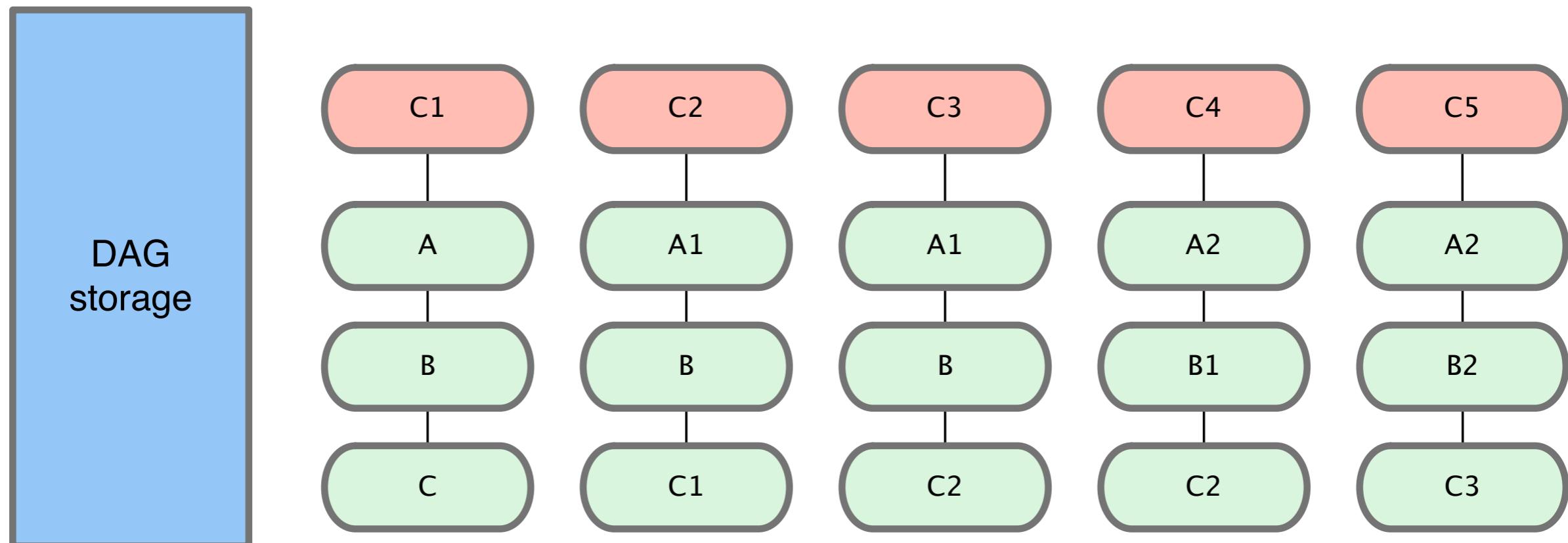
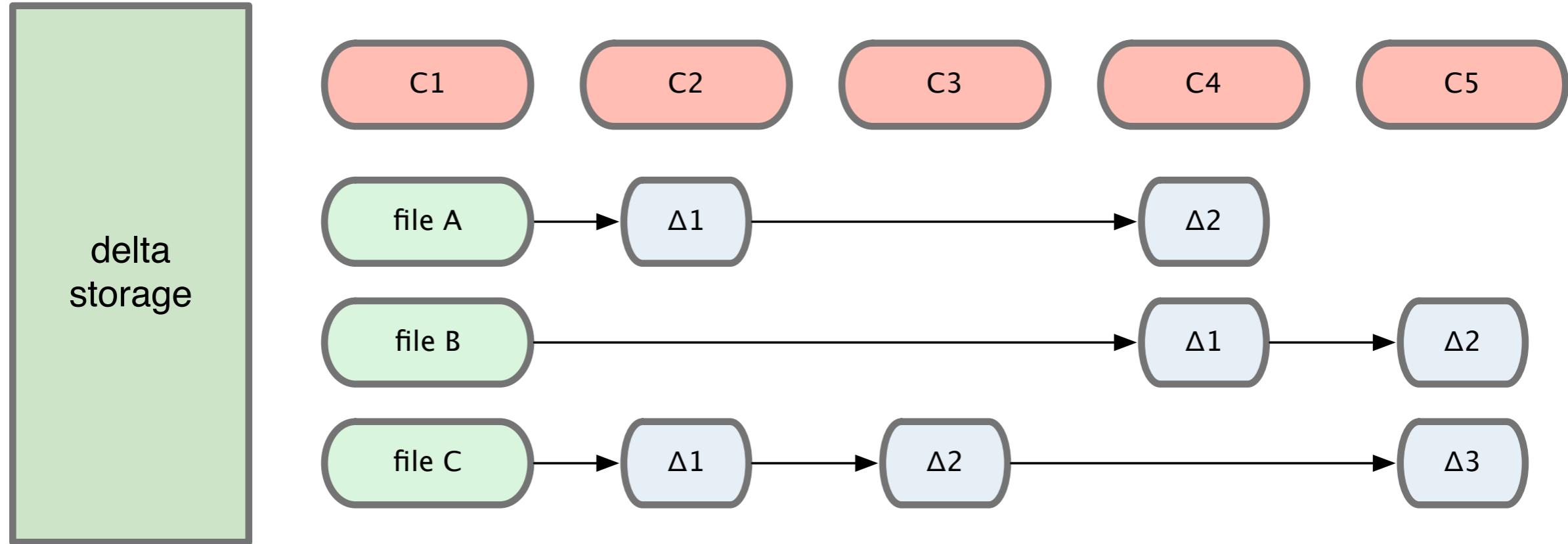
source control taxonomy



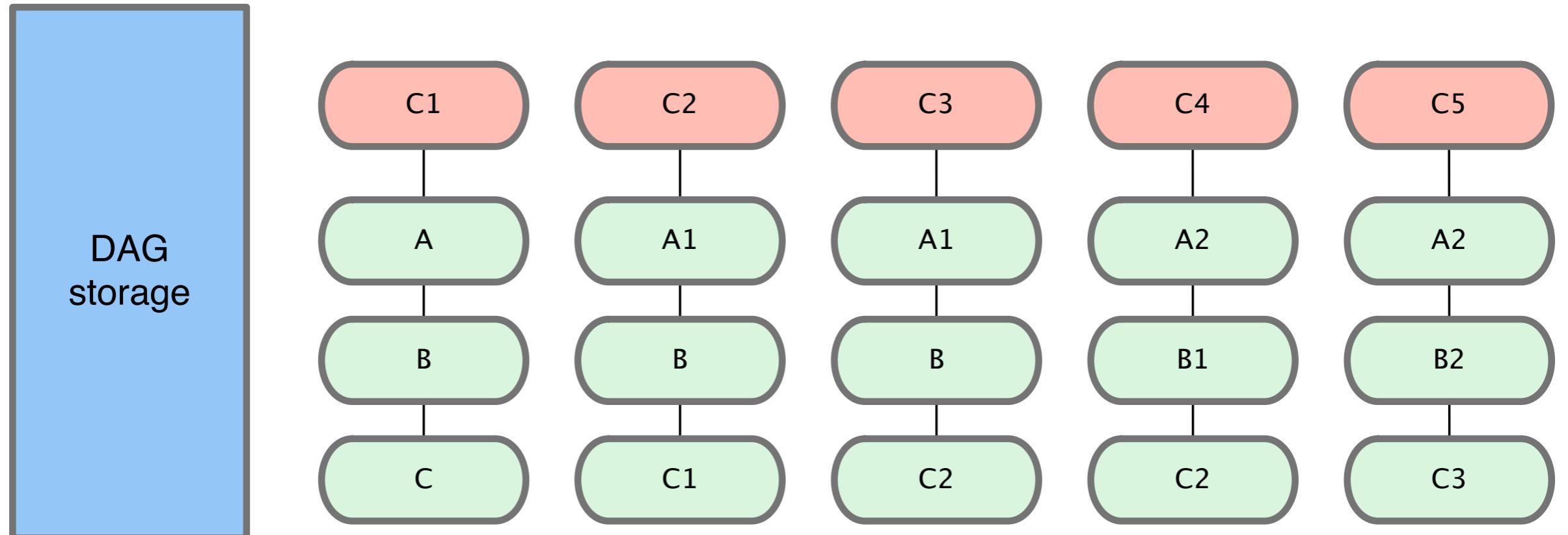
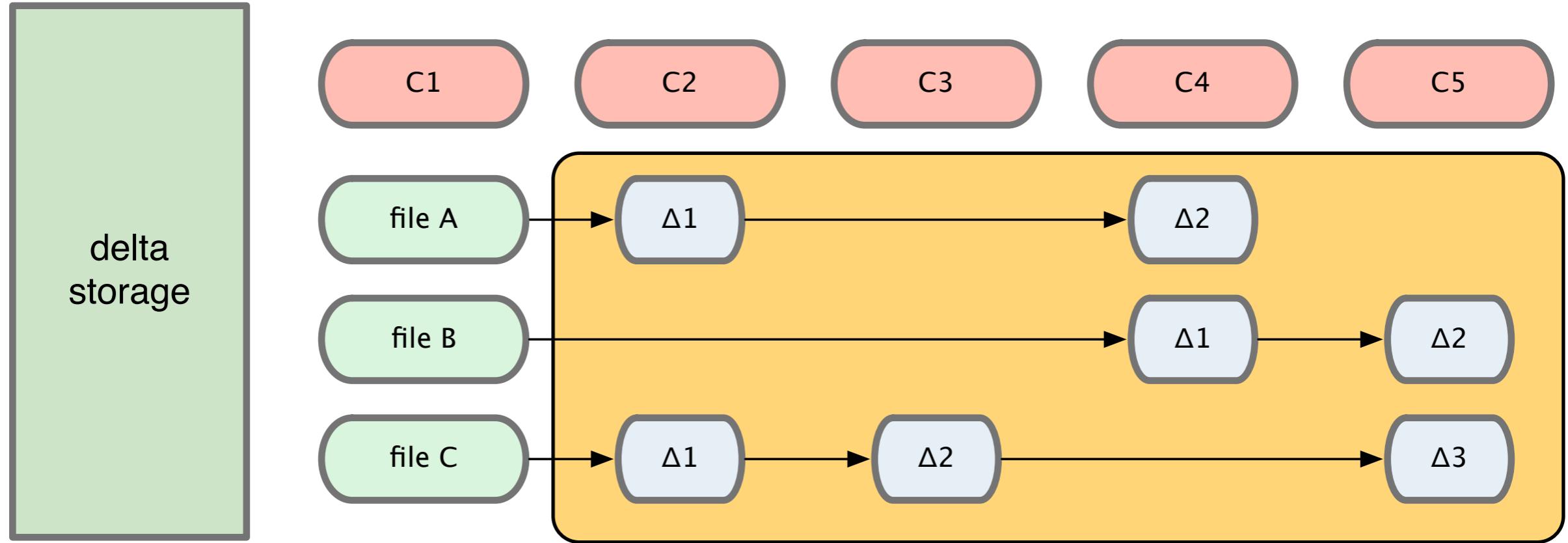
source control taxonomy



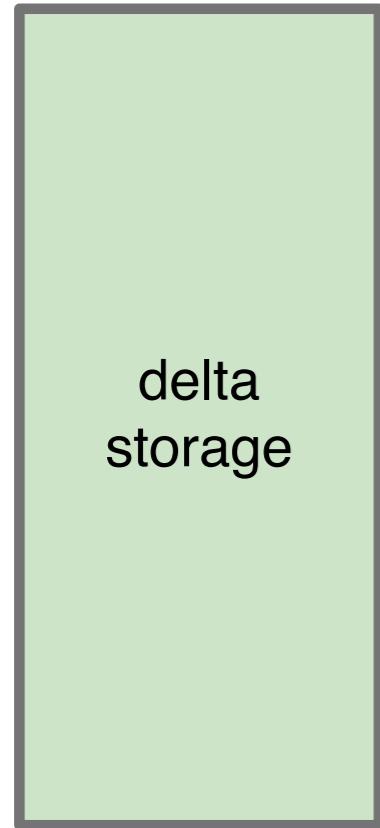
source control taxonomy



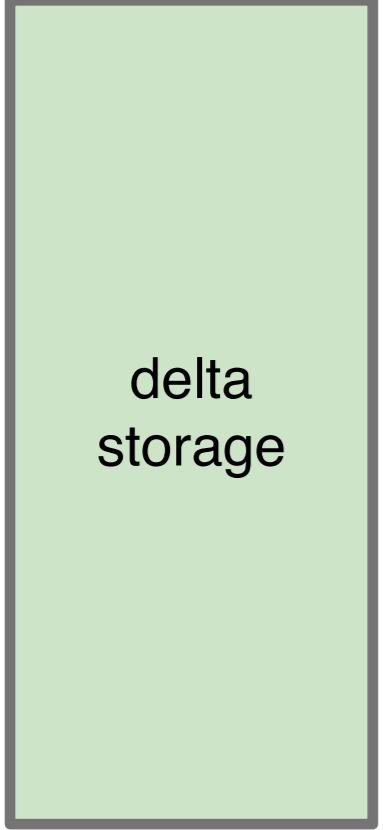
source control taxonomy



source control taxonomy



source control taxonomy

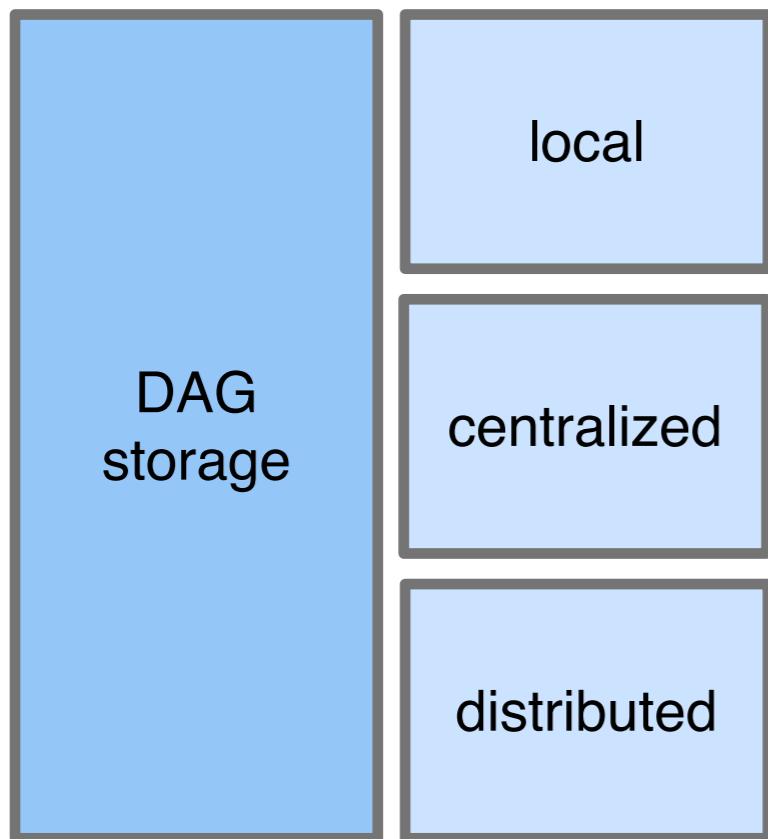
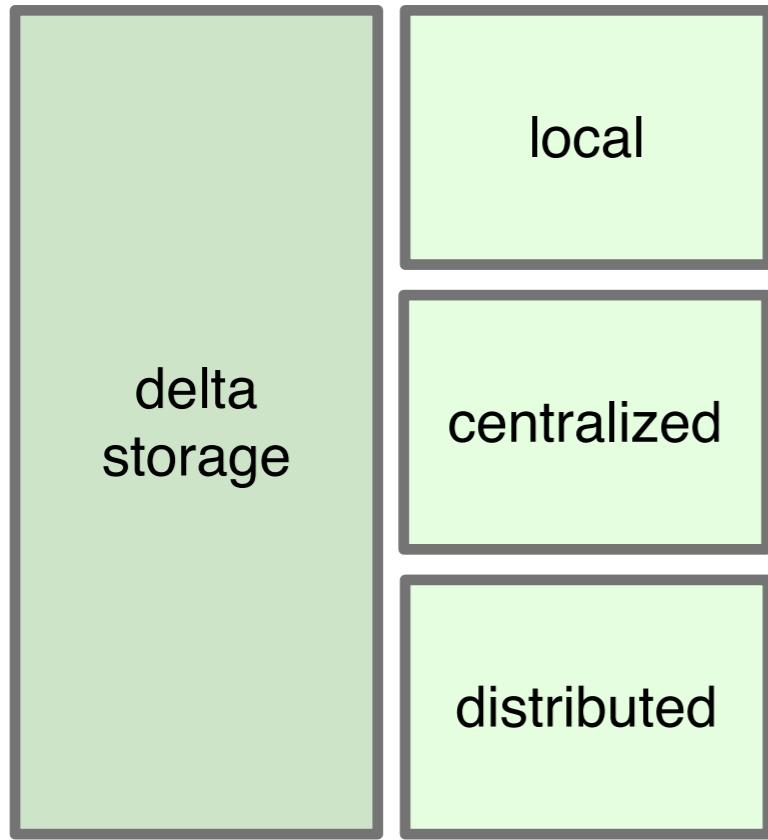


delta
storage

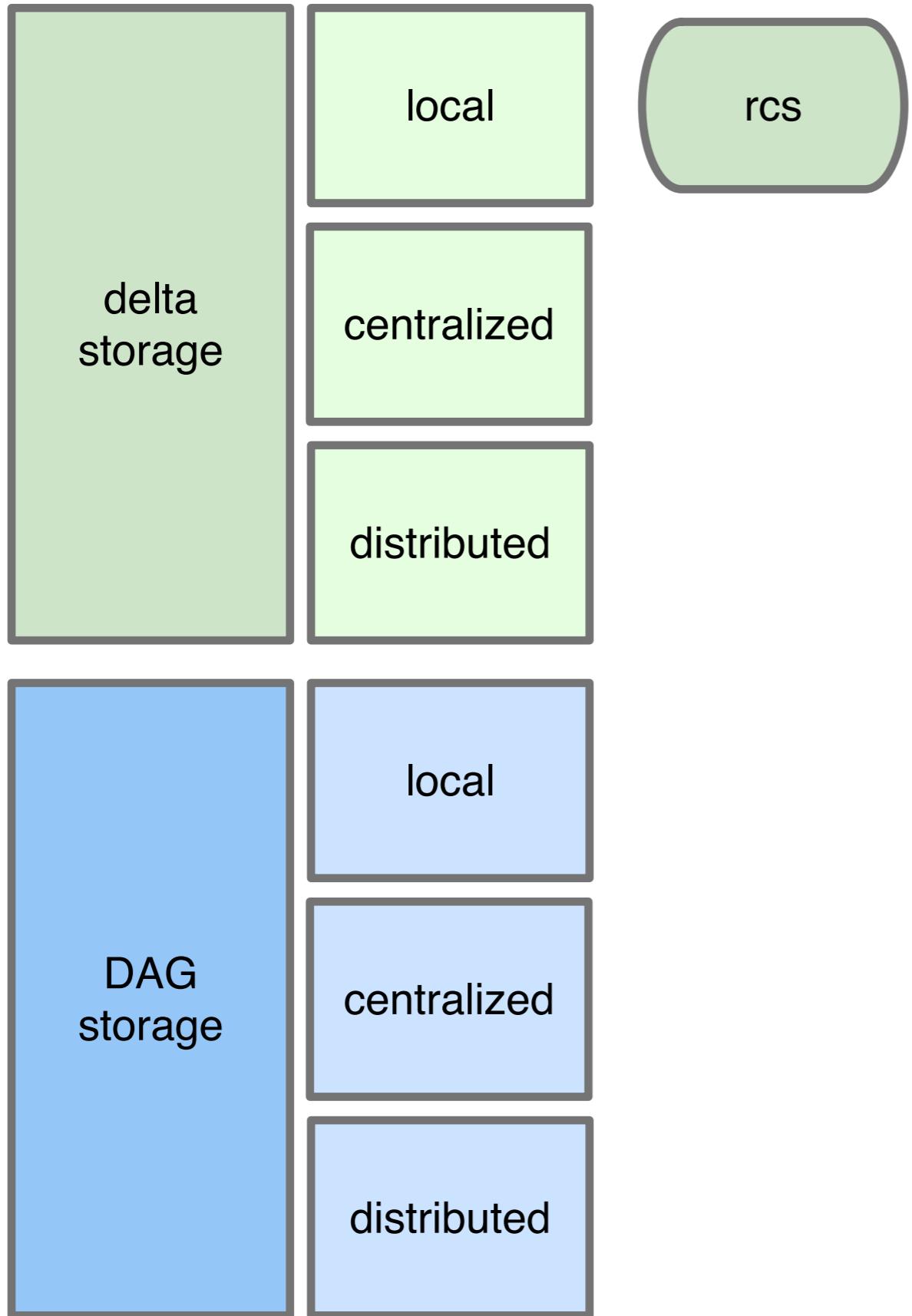


DAG
storage

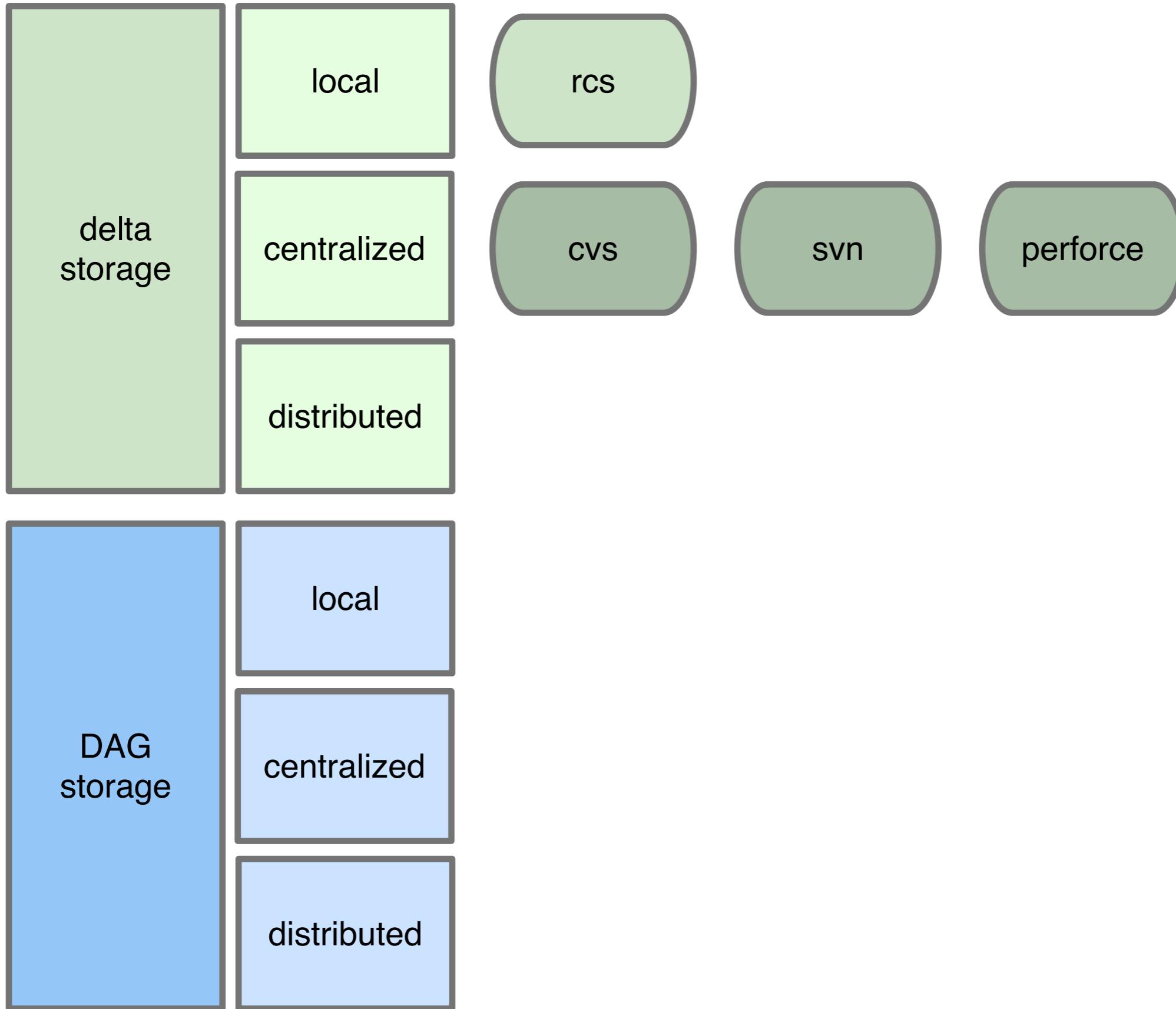
source control taxonomy



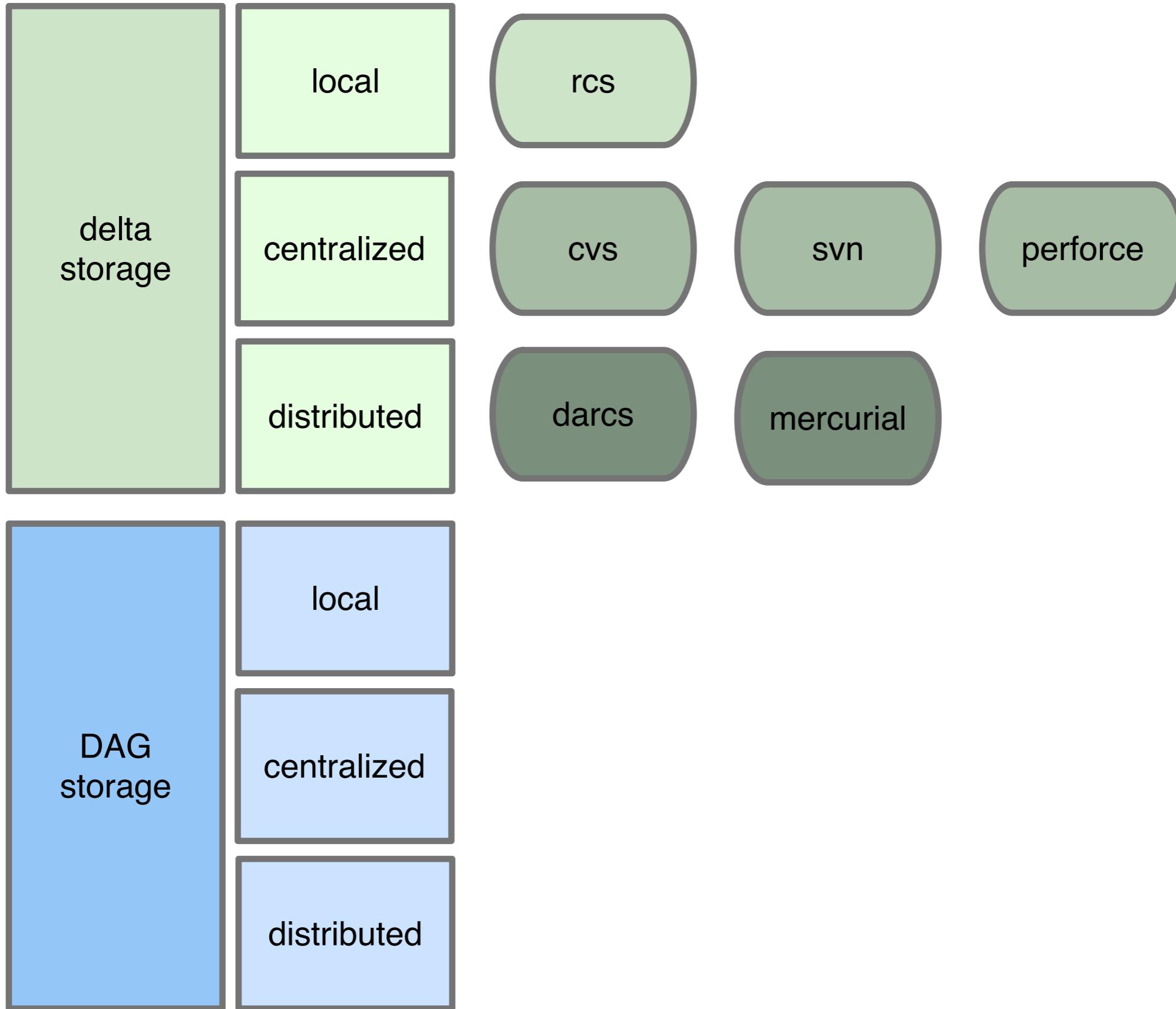
source control taxonomy



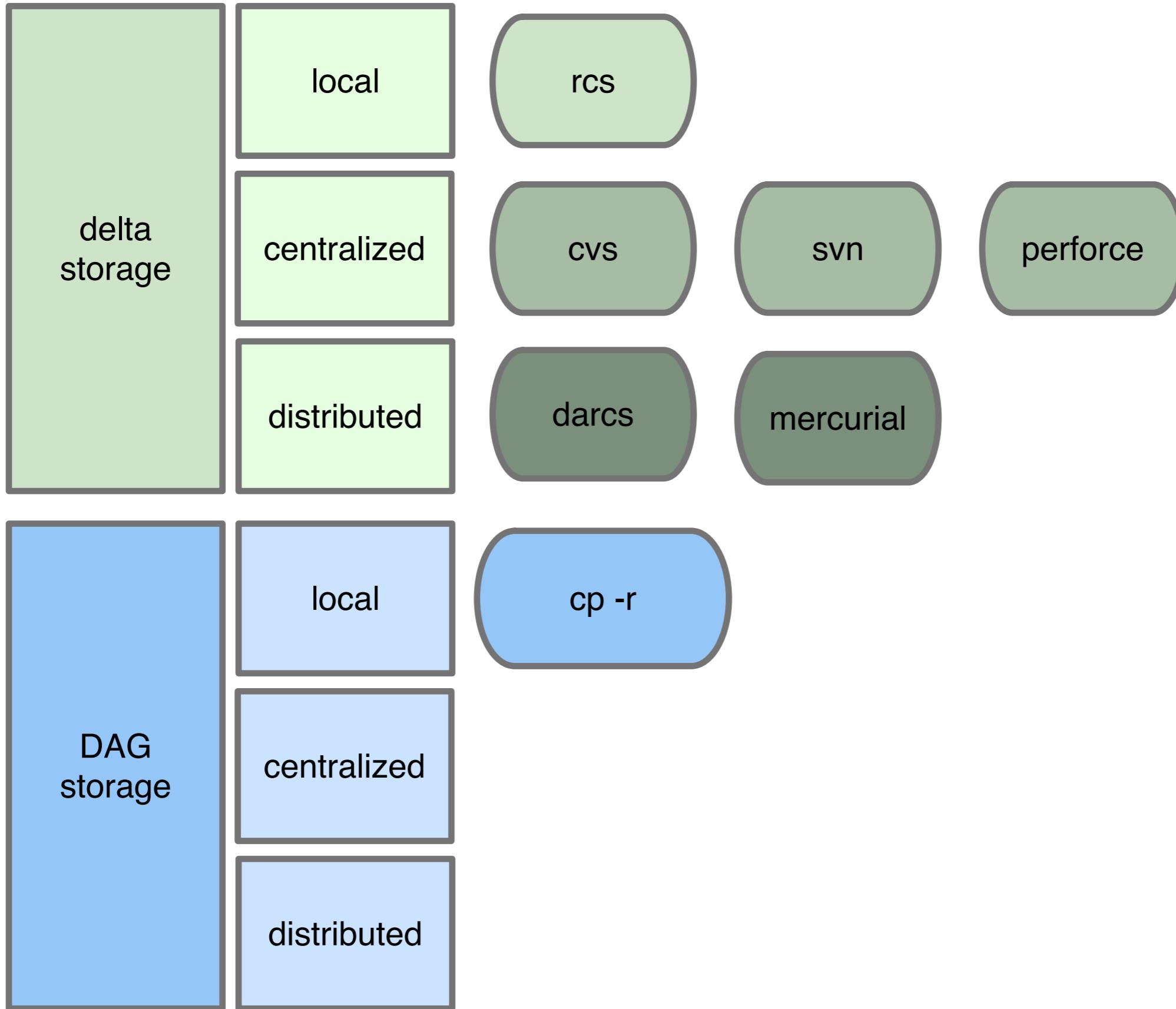
source control taxonomy



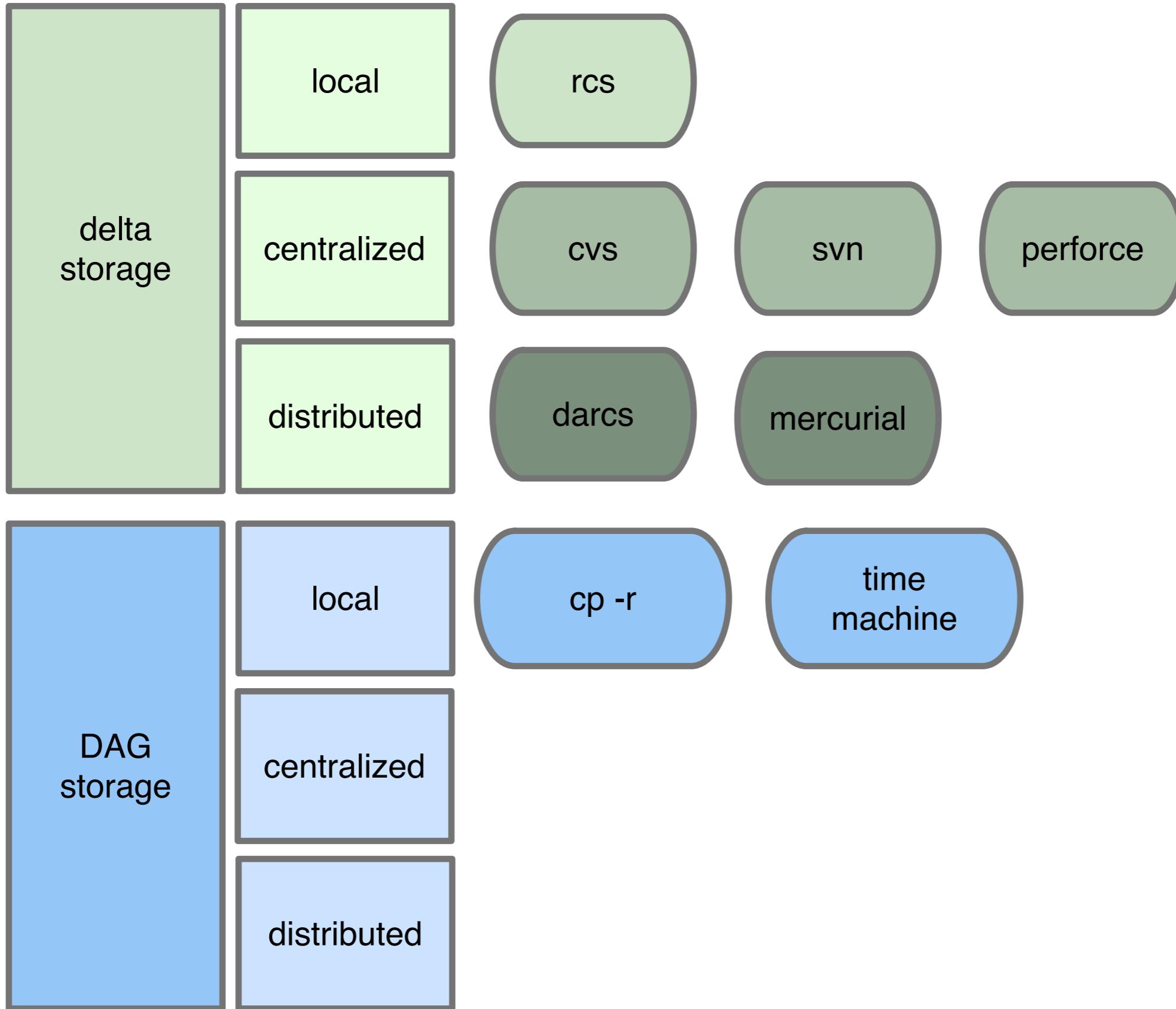
source control taxonomy



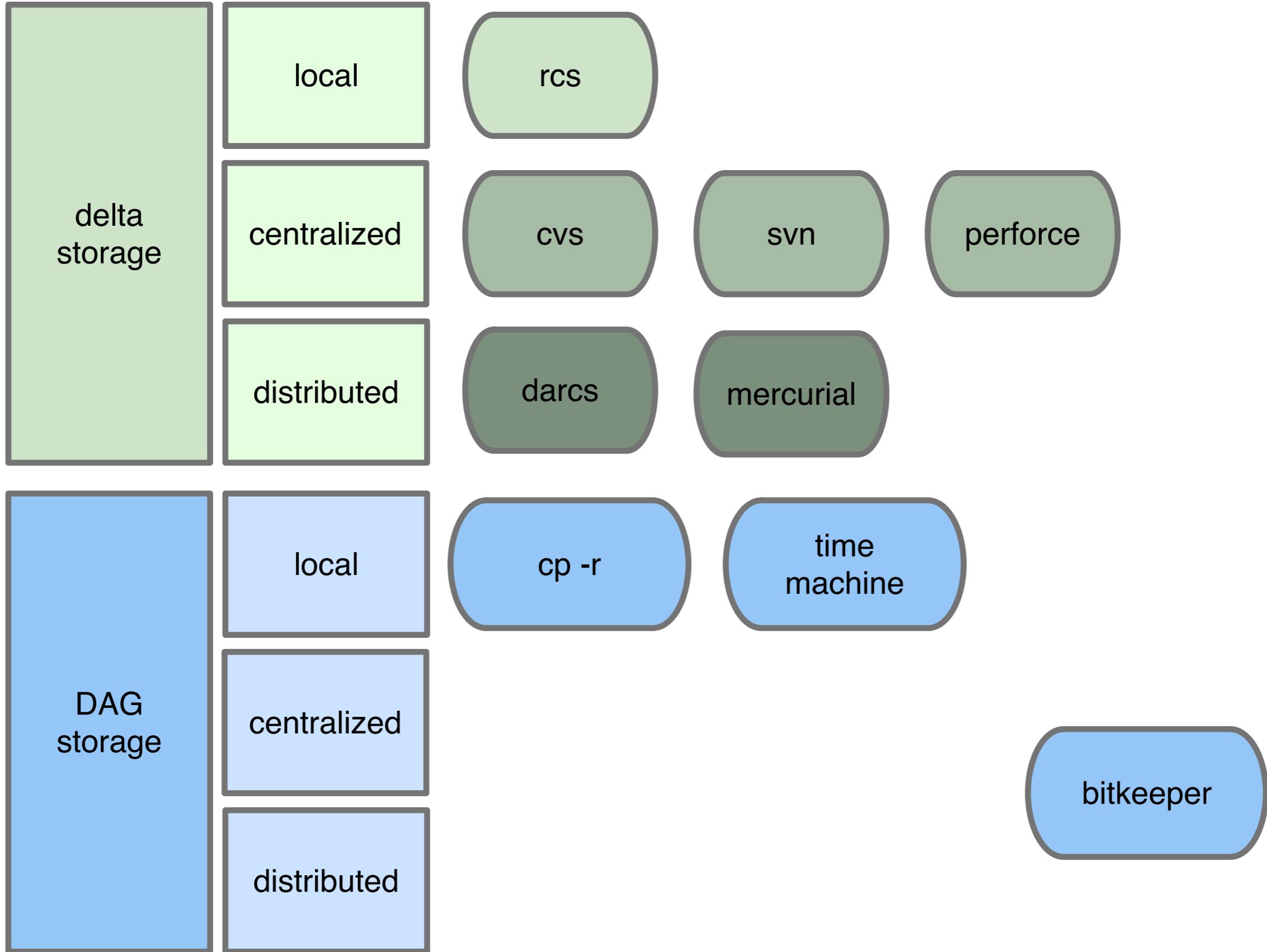
source control taxonomy



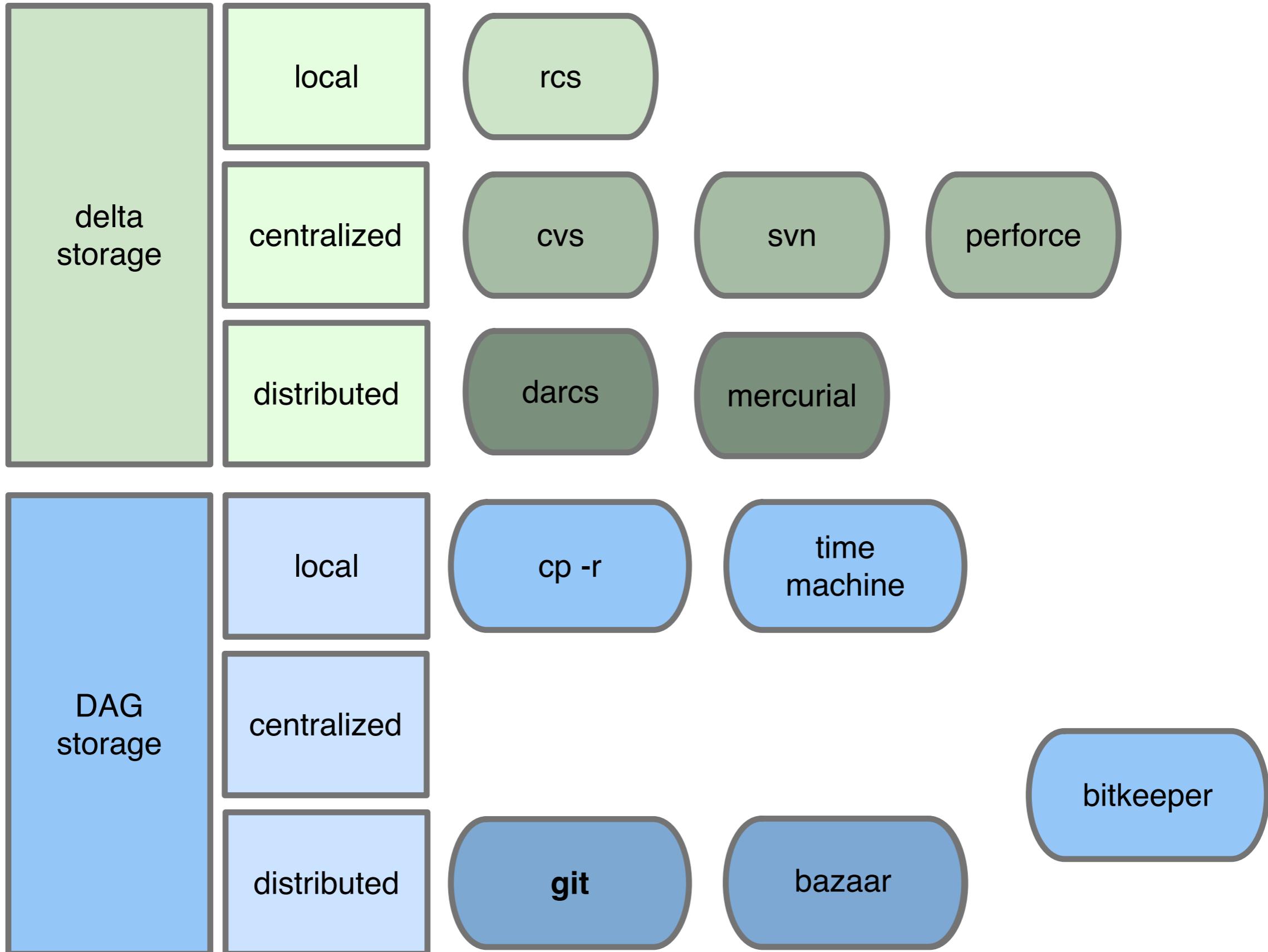
source control taxonomy



source control taxonomy



source control taxonomy





How Does Git Work?

?



**How Does
Git Work?**

10%
4 min

2m

This Slide
Who is Scott?

75%
35 min

20m

How Does Git Work?

15%
6 min

4m

2m

How Do I Use Git?
How Do I Deploy with Git?
Where Can I Learn More?

12m



git

git directory

git directory

.git

git directory

.git

GIT_DIR

Git Directory

config file

hooks

index

object database

references

object database

object database

content

object database

content

```
new_content = type + ' ' + content.size + \  
             + content
```

object database

content “header”

```
new_content = type + ' ' + content.size + \0
               + content
```

object database

content

```
new_content = type + ' ' + content.size + \  
             + content
```

```
sha = Digest::SHA1.hexdigest(new_content)
```

object database

content

```
new_content = type + ' ' + content.size + \  
             + content
```

```
sha = Digest::SHA1.hexdigest(new_content)
```

“824aed035c0aa75d64c...”

object database

content

```
new_content = type + ' ' + content.size + \  
             + content
```

```
sha = Digest::SHA1.hexdigest(new_content)
```

“824aed035c0aa75d64c...”

```
compressed = zlib::deflate(new_content)
```

object database

content

```
new_content = type + ' ' + content.size + \  
             + content
```

```
sha = Digest::SHA1.hexdigest(new_content)
```

“824aed035c0aa75d64c...”

```
compressed = zlib::deflate(new_content)
```

```
path = ".git/objects/82/4aed035c0aa75d64c..."
```

object database

content

```
new_content = type + ' ' + content.size + \  
             + content
```

```
sha = Digest::SHA1.hexdigest(new_content)
```

“824aed035c0aa75d64c...”

```
compressed = zlib::deflate(new_content)
```

```
path = ".git/objects/82/4aed035c0aa75d64c..."
```

```
File.open(path, 'w') { |f| f.write(compressed) }
```

object database

content

```
new_content = type + ' ' + content.size + \  
             + content
```

“loose” format

```
sha = Digest::SHA1.hexdigest(new_content)
```

“824aed035c0aa75d64c...”

```
compressed = zlib::deflate(new_content)
```

```
path = ".git/objects/82/4aed035c0aa75d64c..."
```

```
File.open(path, 'w') {|f| f.write(compressed)}
```

object database

git gc

object database

git gc

same file with minor differences

object database

git gc

same file with minor differences

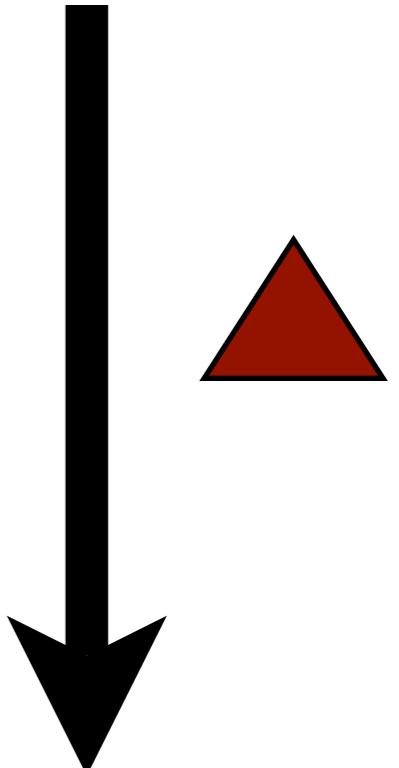
```
.git/objects/82/4aed035c0aa75d64c...
.git/objects/1d/c9cbb76cbb80fce1...
.git/objects/63/874f37013c1740acd...
.git/objects/04/fb8aee105e6e445e8...
.git/objects/45/b983be36b73c0788d...
.git/objects/f1/032eed02413a1145c...
```

object database

git gc

same file with minor differences

```
.git/objects/82/4aed035c0aa75d64c...
.git/objects/1d/c9cbb76cbb80fce1...
.git/objects/63/874f37013c1740acd...
.git/objects/04/fb8aee105e6e445e8...
.git/objects/45/b983be36b73c0788d...
.git/objects/f1/032eed02413a1145c...
```



```
.git/objects/pack/pack-999727..9f600.pack
.git/objects/pack/pack-999727..9f600.idx
```

object database

git gc

same file with minor differences

```
.git/objects/82/4aed035c0aa75d64c...
.git/objects/1d/c9cbb76cbb80fce1...
.git/objects/63/874f37013c1740acd...
.git/objects/04/fb8aee105e6e445e8...
.git/objects/45/b983be36b73c0788d...
.git/objects/f1/032eed02413a1145c...
```

```
.git/objects/pack/pack-999727..9f600.pack
.git/objects/pack/pack-999727..9f600.idx
```

object database

git gc

same file with minor differences

.git/objects/82/4aed0...
“packed” format
.git/objects/1d/99c...
.git/objects/63/874f37013c1740acd...
.git/objects/04/fb8aee105e6e445e8...
.git/objects/45/b983be36b73c0788d...
.git/objects/f1/032eed02413a1145c...

.git/objects/pack/pack-999727..9f600.pack
.git/objects/pack/pack-999727..9f600.idx

object database

all git objects are stored this way

object database

```
new_content = type + ' ' + content.size + \  
             + content
```

4 types of git objects

object database

blob

object database

blob

tree

object database

blob

tree

commit

object database

blob

tree

commit

tag

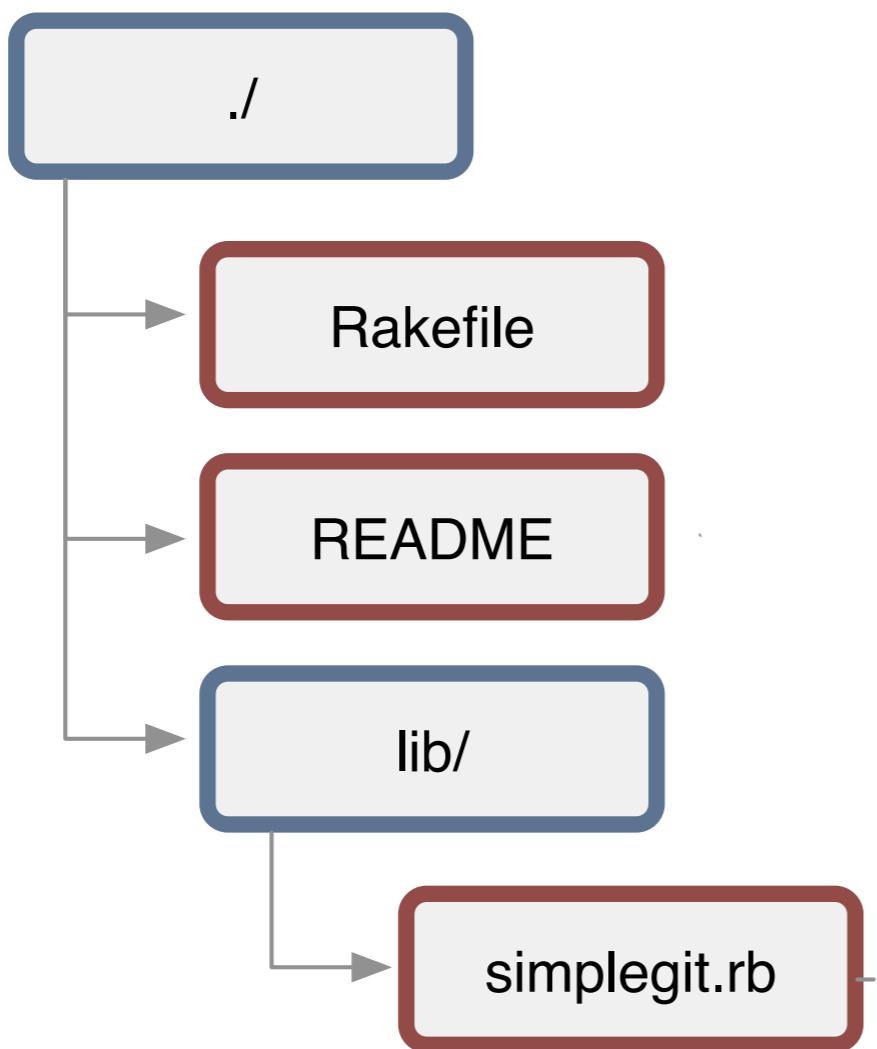
object database

blob

object database

blob

Working Directory



object database

blob : a906cb

zlib::deflate

blob [content size]\0

SimpleGit Ruby Library

=====

This library calls git commands and returns the output.

Author : Scott Chacon

object database

blob : a906cb

zlib::deflate

blob [content size]\0

SimpleGit Ruby Library

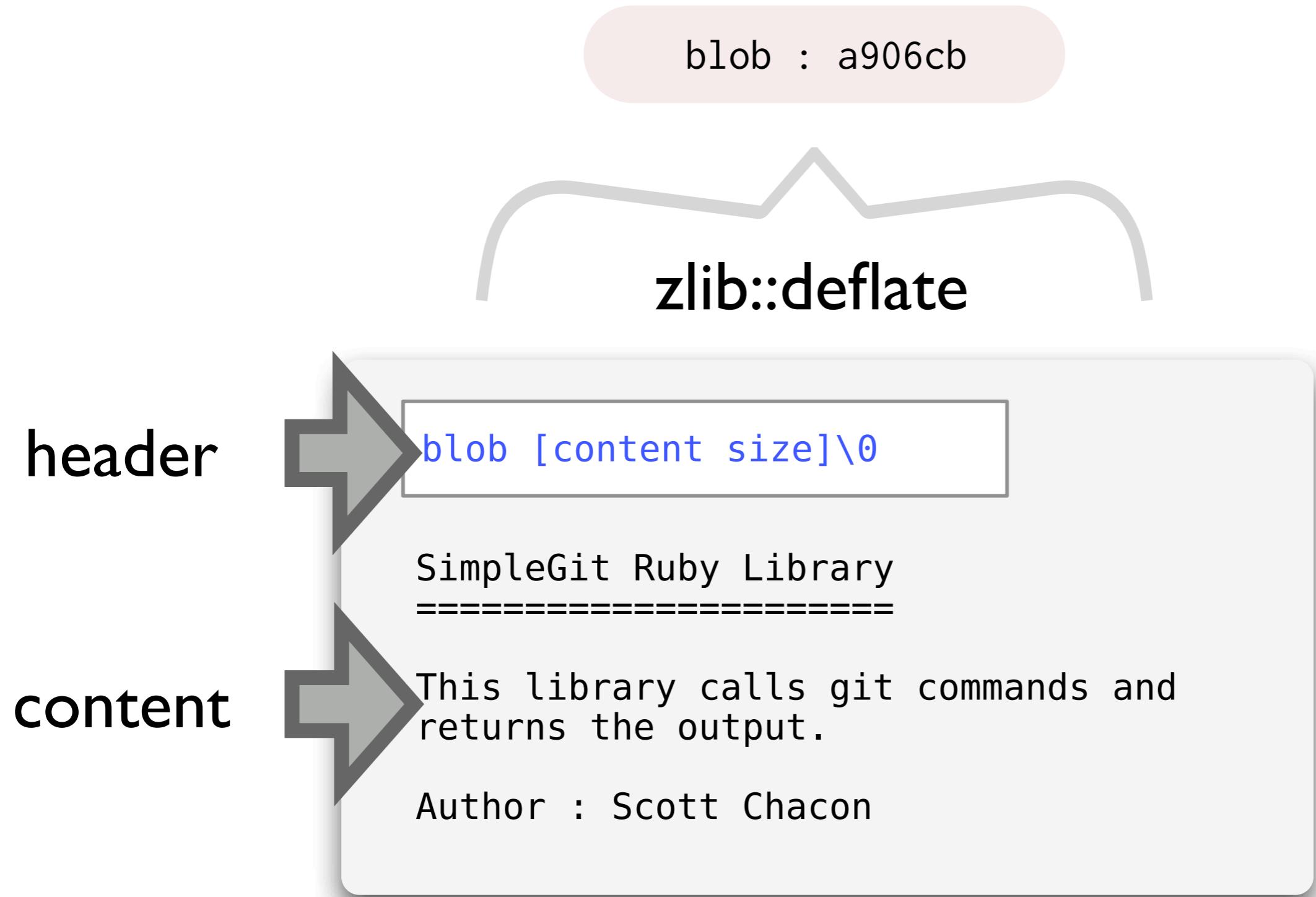
=====

This library calls git commands and
returns the output.

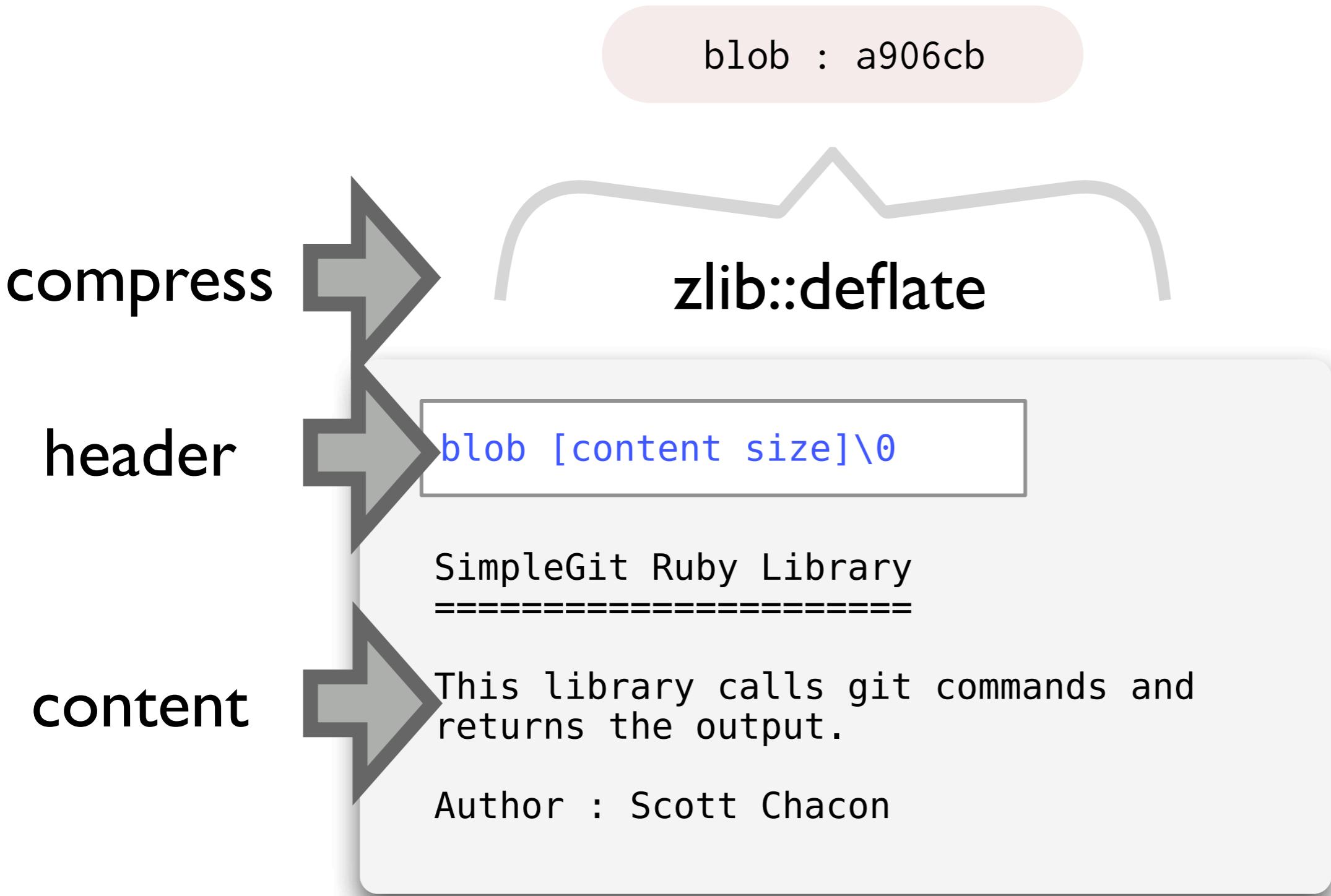
Author : Scott Chacon

content

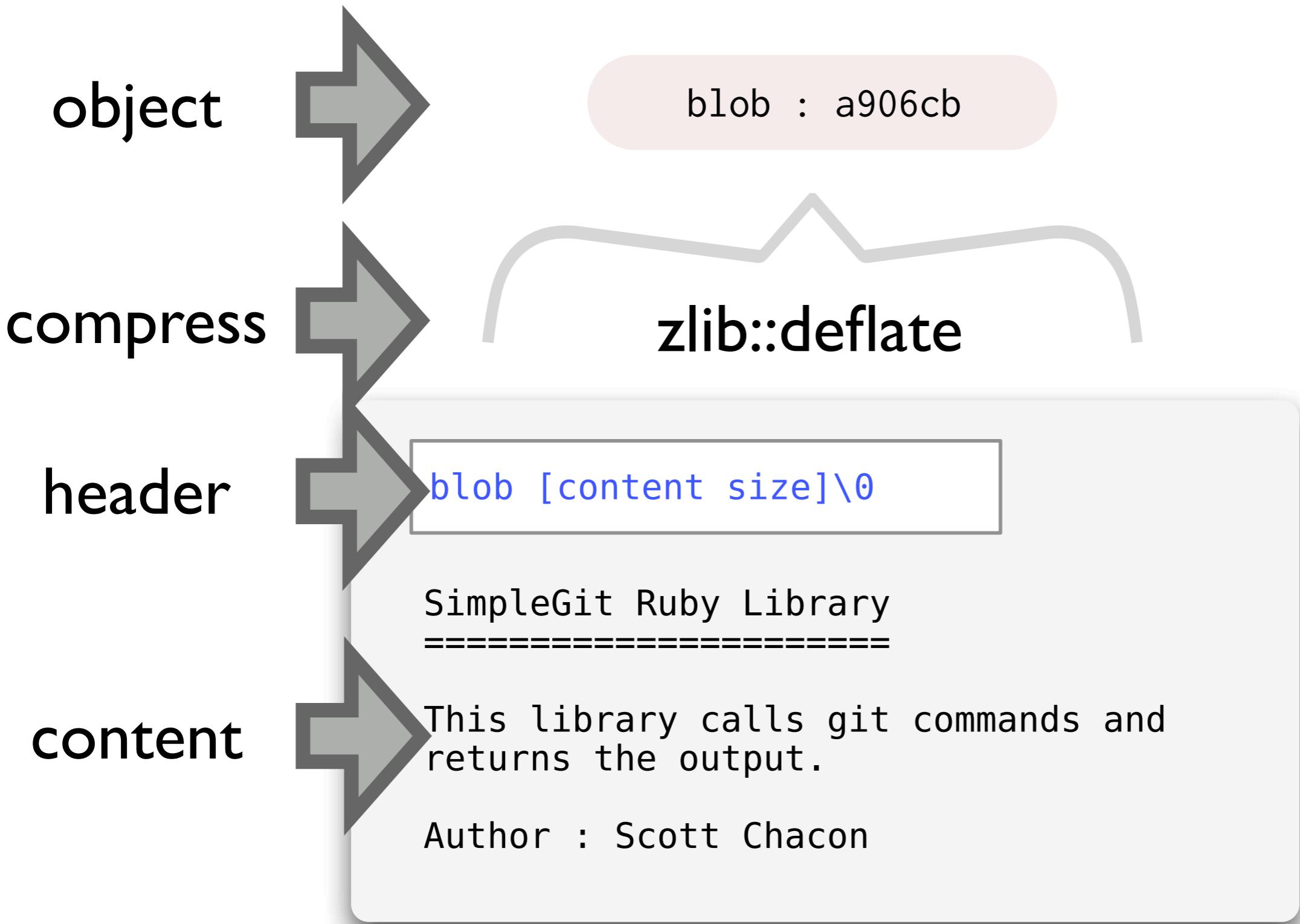
object database



object database



object database



object database

blob

tree

commit

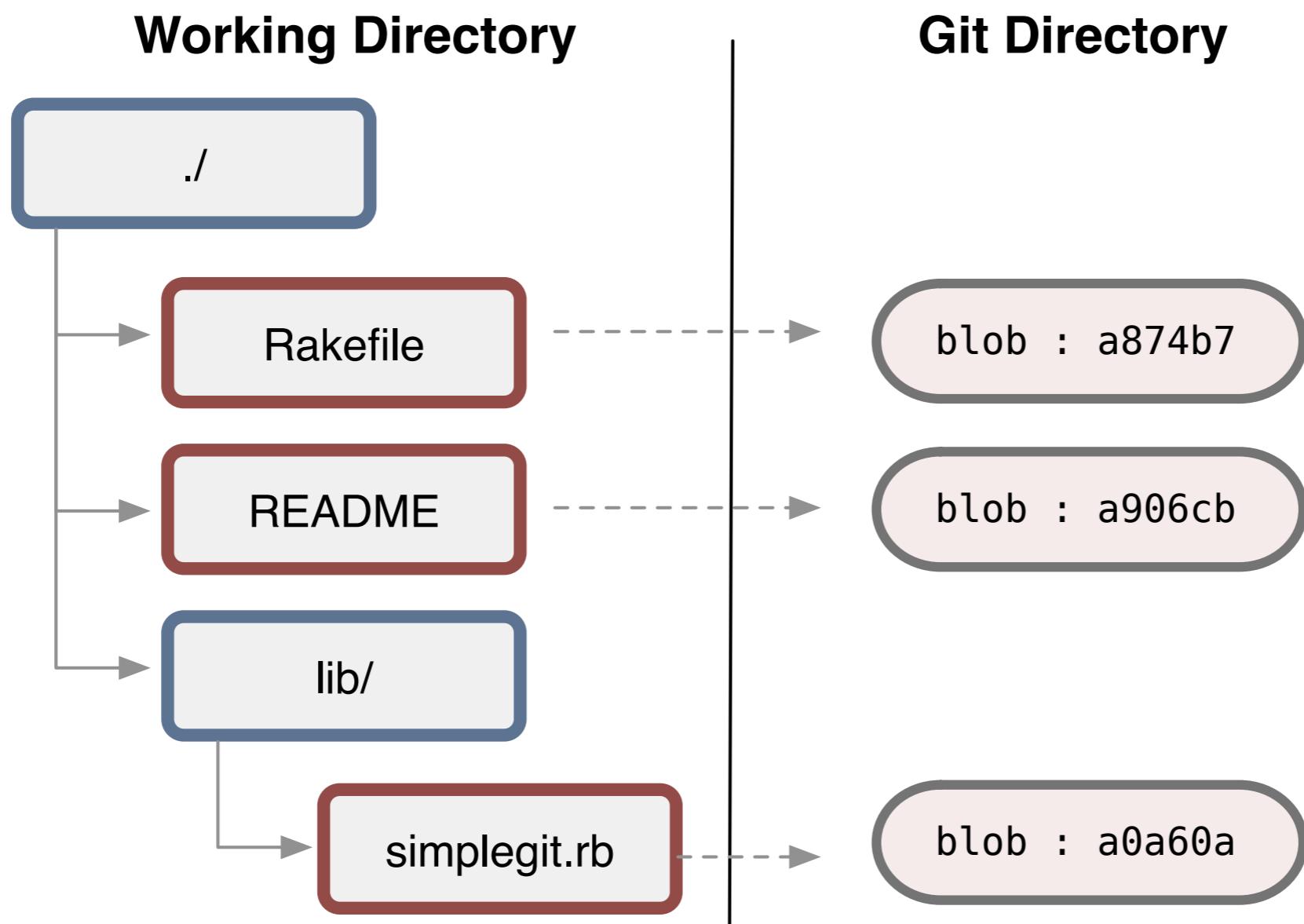
tag

object database

tree

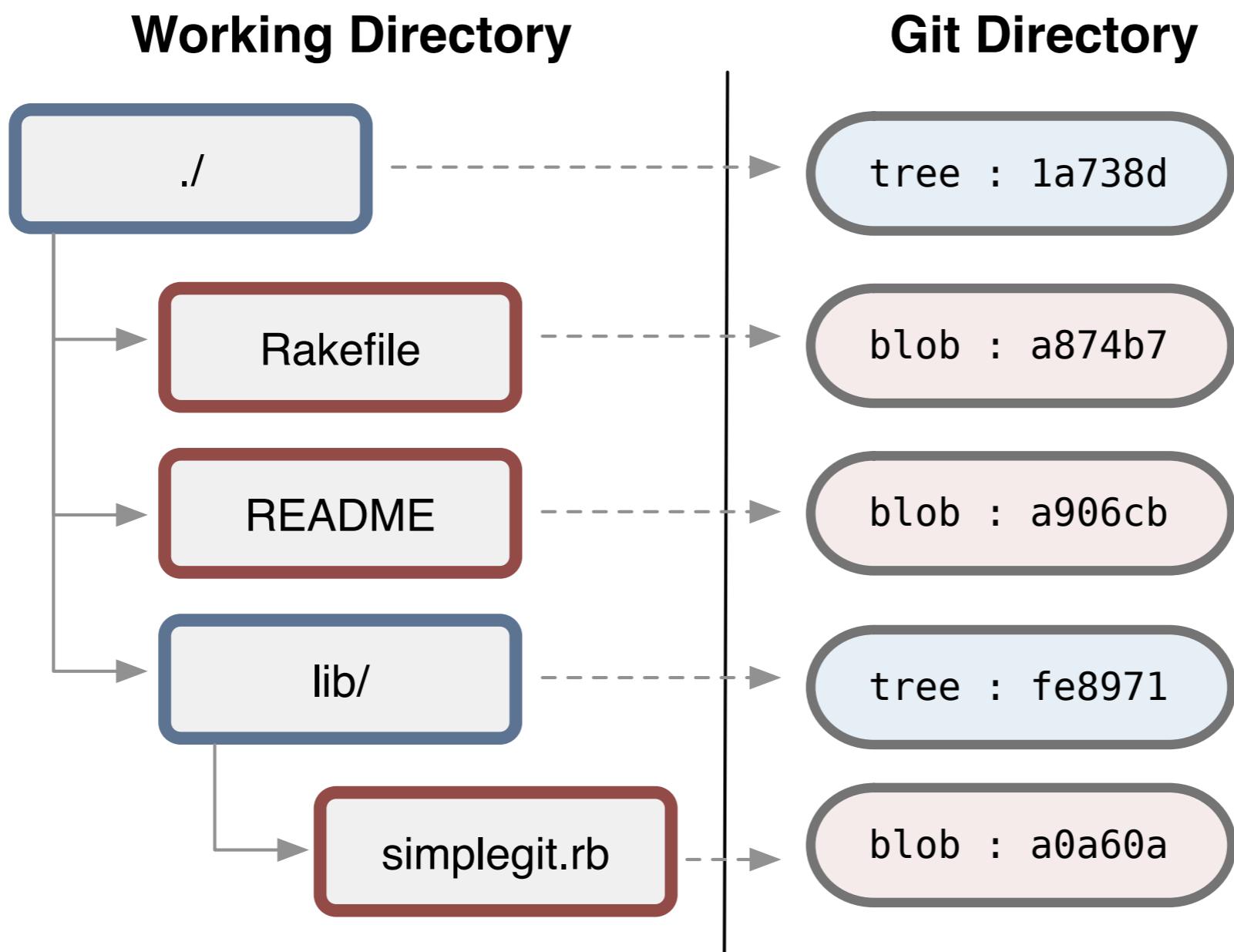
object database

tree



object database

tree



object database

tree : 1a738d

zlib::deflate

tree [content size]\0

100644	blob	a906cb	README
100644	blob	a874b7	Rakefile
040000	tree	fe8971	lib

object database

tree : 1a738d

zlib::deflate

tree [content size]\0

100644 blob a906cb

100644 blob a874b7

040000 tree fe8971

README

Rakefile

lib

“inode” info

filename

object database

tree : 1a738d

zlib::deflate

tree [content size]\0

100644	blob	a906cb	README
100644	blob	a874b7	Rakefile
040000	tree	fe8971	lib

mode type “block pointer”

object database

blob

tree

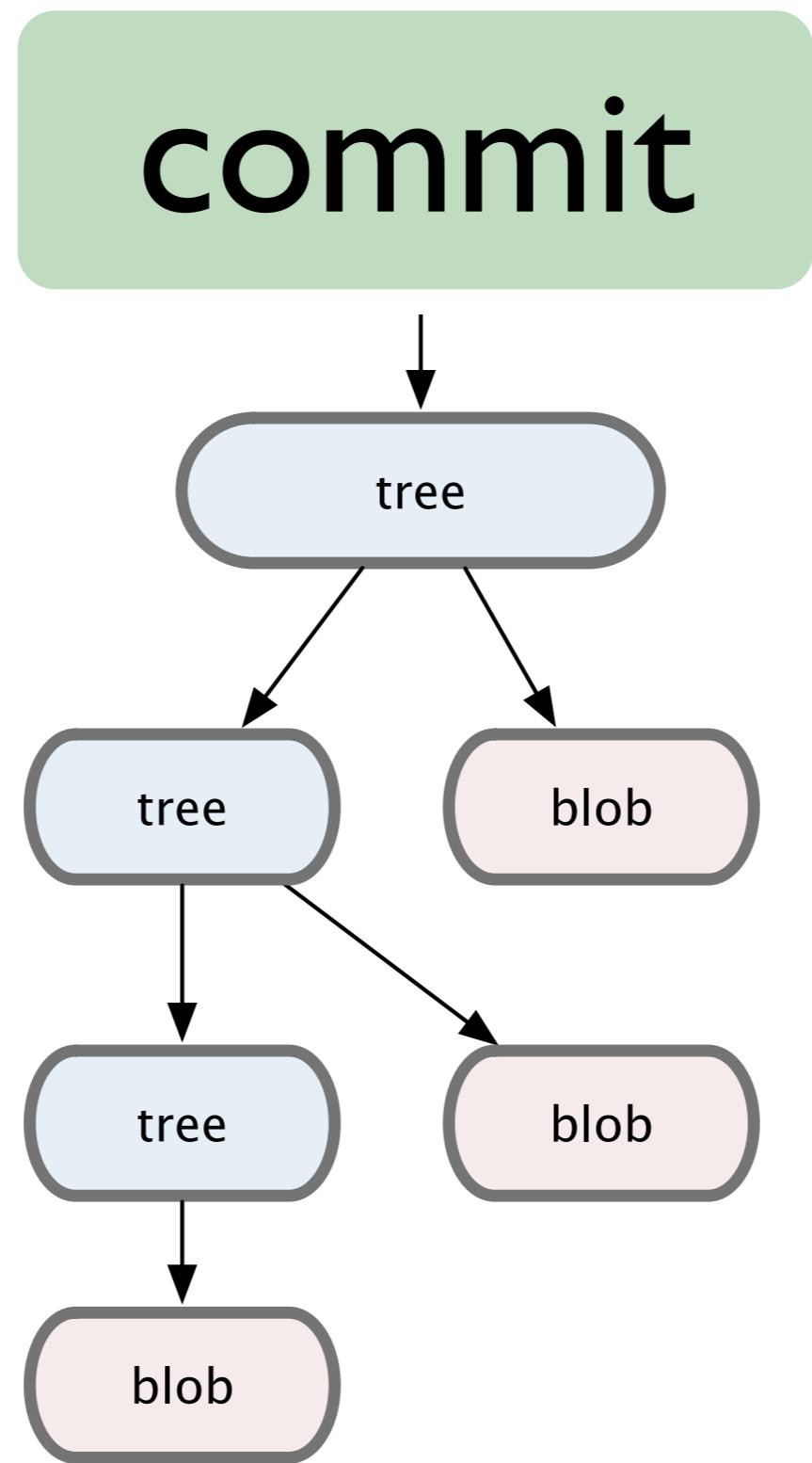
commit

tag

object database

commit

object database



object database

commit : e1b3ec

zlib::deflate

```
commit [content size]\0
```

```
tree e1b3ec
parent a11bef
author Scott Chacon
    <schacon@gmail.com> 1205624433
committer Scott Chacon
    <schacon@gmail.com> 1205624433
```

my second commit, which is better than
the first

object database

commit : e1b3ec

zlib::deflate

commit [content size]\0

```
tree e1b3ec
parent a11bef
author Scott Chacon
<schacon@gmail.com> 1205624433
committer Scott Chacon
<schacon@gmail.com> 1205624433
```

my second commit, which is better than
the first

object database

commit : e1b3ec

zlib::deflate

commit [content size]\0

tree e1b3ec
parent a11bef
author Scott Chacon
<schacon@gmail.com> 1205624433
committer Scott Chacon
<schacon@gmail.com> 1205624433

my second commit, which is better than
the first

object database

commit : e1b3ec

zlib::deflate

commit [content size]\0

tree e1b3ec
parent a11bef
author Scott Chacon
<schacon@gmail.com> 1205624433
committer Scott Chacon
<schacon@gmail.com> 1205624433



my second commit, which is better than
the first

object database

commit : e1b3ec

zlib::deflate

commit [content size]\0

tree e1b3ec

parent a11bef

author Scott Chacon

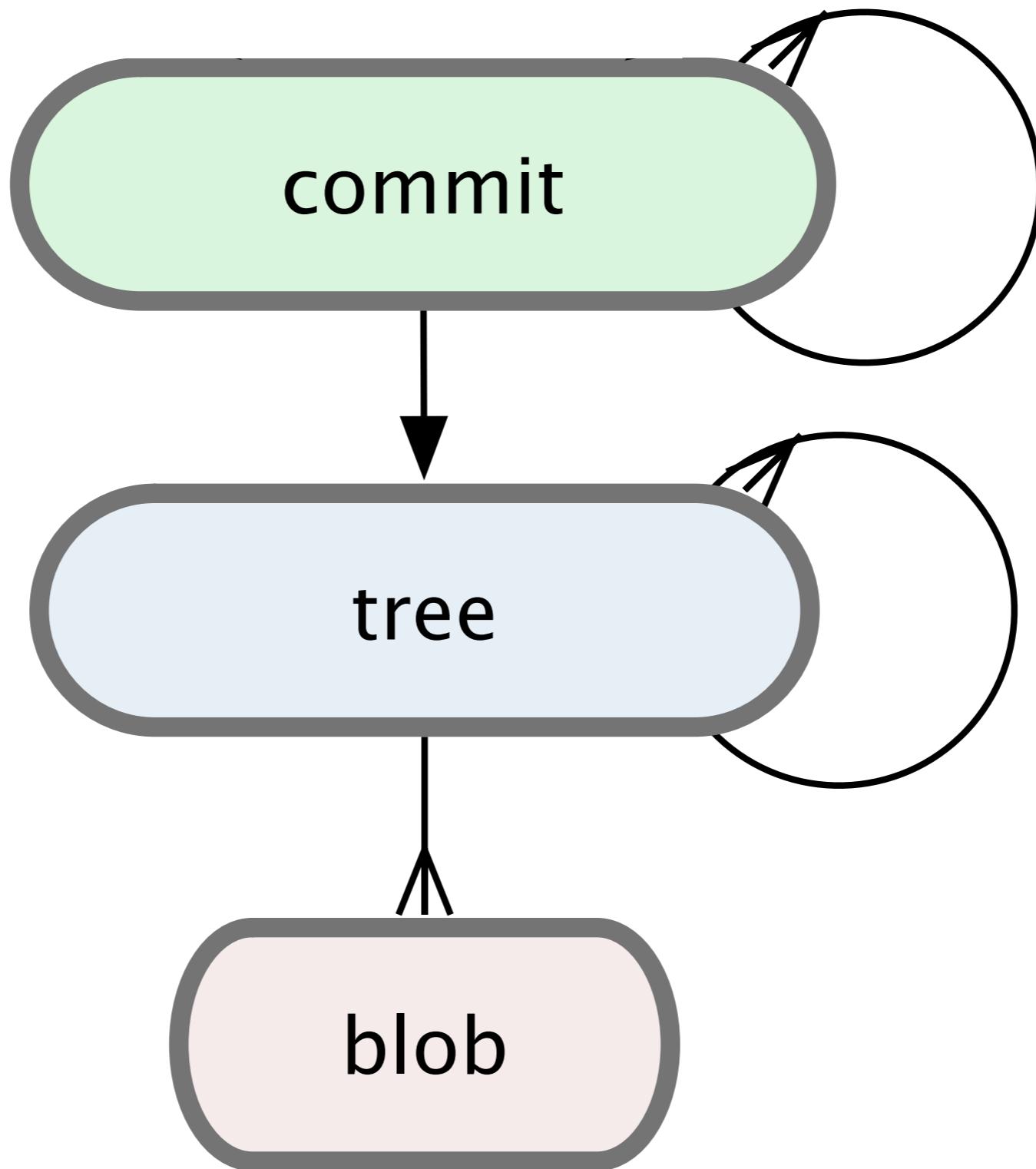
<schacon@gmail.com> 1205624433

committer Scott Chacon

<schacon@gmail.com> 1205624433

my second commit, which is better than
the first

object database



object database

blob

tree

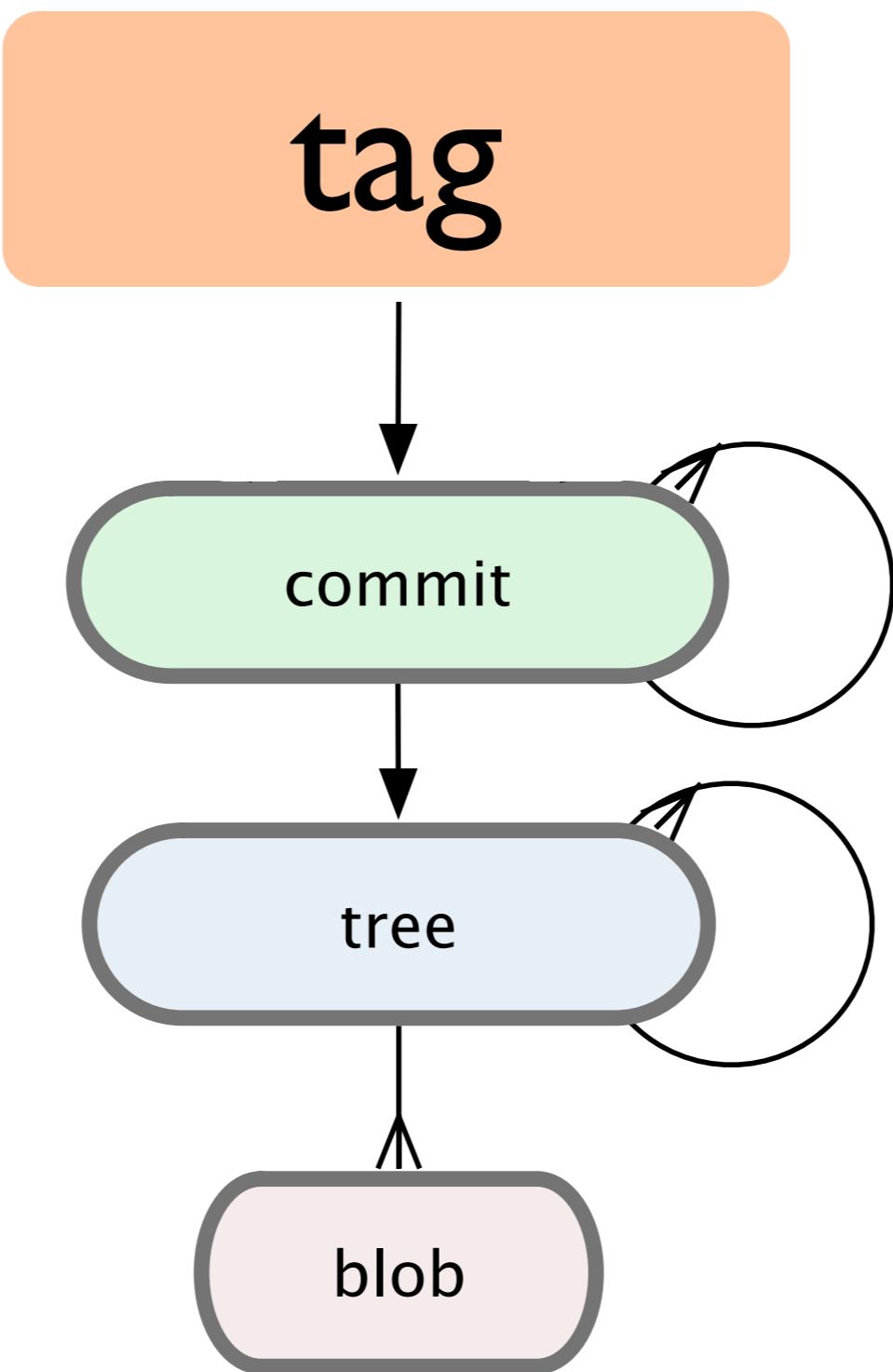
commit

tag

object database

tag

object database



object database

tag : 0c819c

zlib::deflate

tag [content size]\0

```
object 0576fa
type commit
tag v0.1
tagger Scott Chacon
    <schacon@gmail.com> 1205624655
```

this is my v0.1 tag

object database

tag : 0c819c

zlib::deflate

tag [content size]\0

object 0576fa
type commit
tag v0.1
tagger Scott Chacon
<schacon@gmail.com> 1205624655

this is my v0.1 tag

object database

tag : 0c819c

zlib::deflate

tag [content size]\0

object 0576fa

type commit

tag v0.1

tagger Scott Chacon

<schacon@gmail.com> 1205624655

this is my v0.1 tag



object database

tag : 0c819c

zlib::deflate

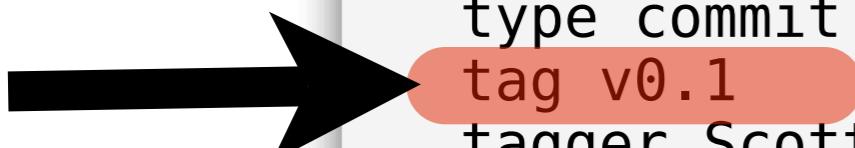
tag [content size]\0

object 0576fa
type commit

tag v0.1

tagger Scott Chacon
<schacon@gmail.com> 1205624655

this is my v0.1 tag



object database

tag : 0c819c

zlib::deflate

tag [content size]\0

object 0576fa
type commit

tag v0.1

tagger Scott Chacon
<schacon@gmail.com> 1205624655

this is my v0.1 tag



object database

blob

tree

commit

tag

object database

blob

tree

immutable

commit

tag

object database

blob

tree

immutable

commit

tag

can't be muted

Git Directory

config file

hooks

index

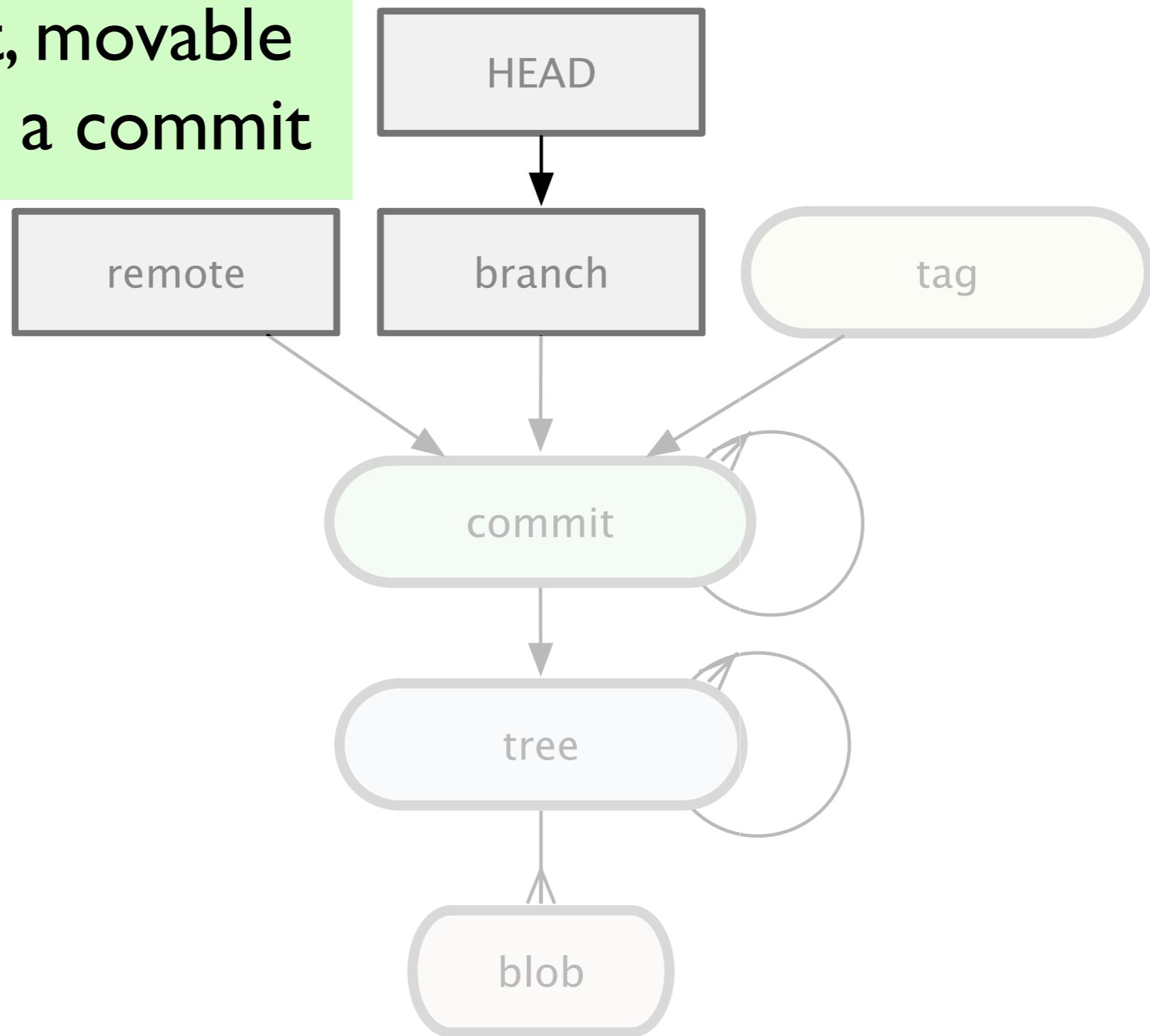
object database

references

references

references

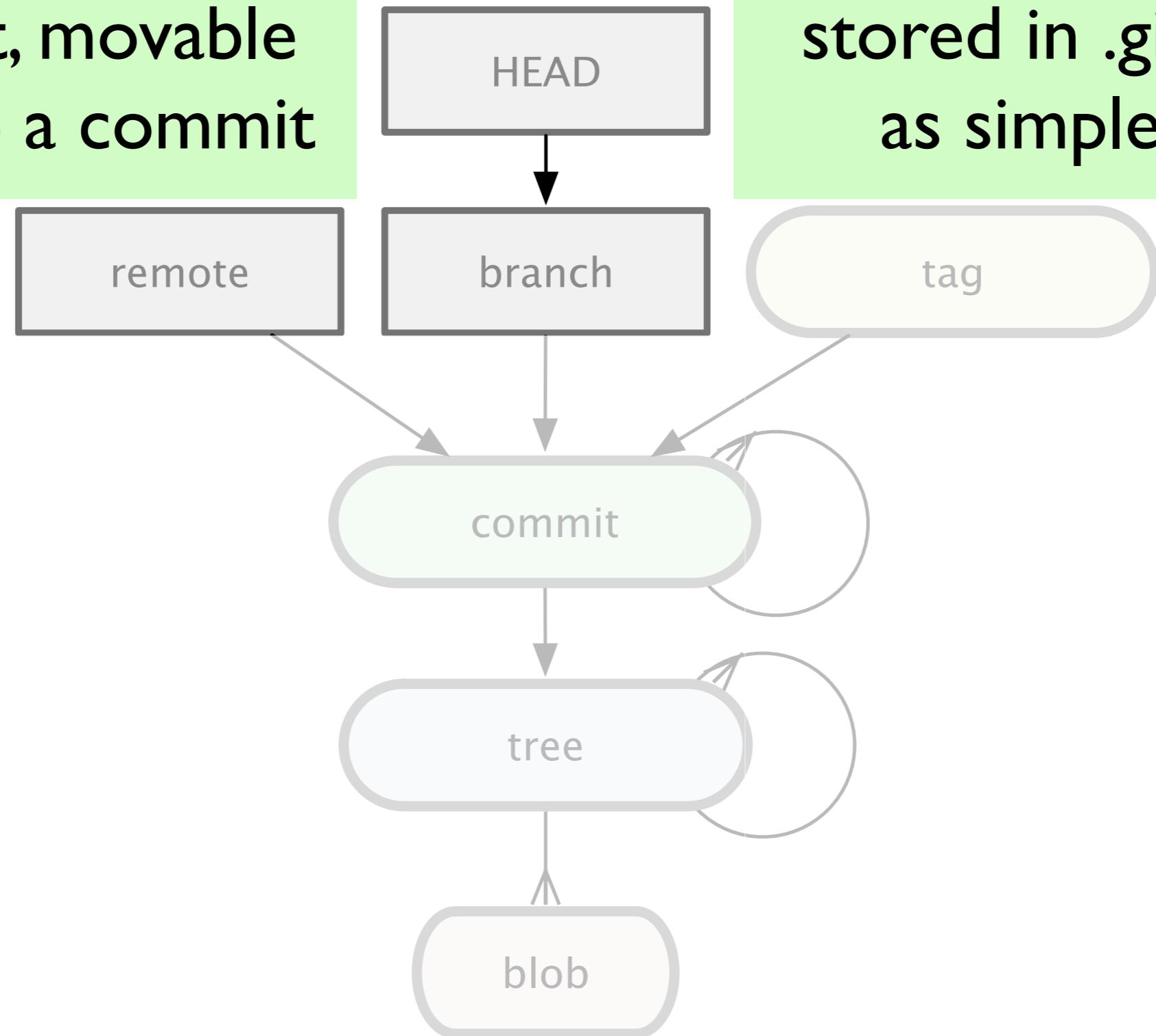
lightweight, movable
pointers to a commit



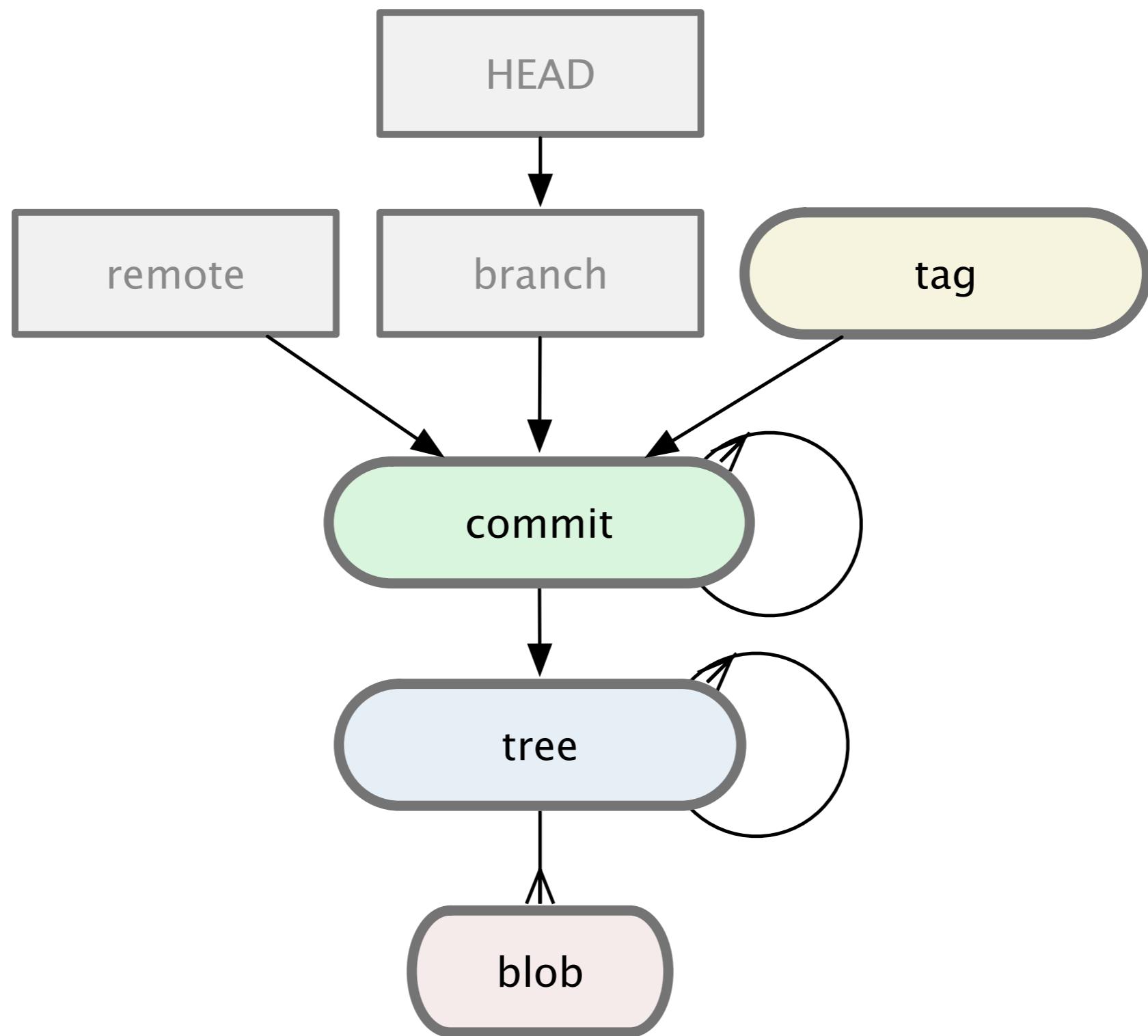
references

lightweight, movable
pointers to a commit

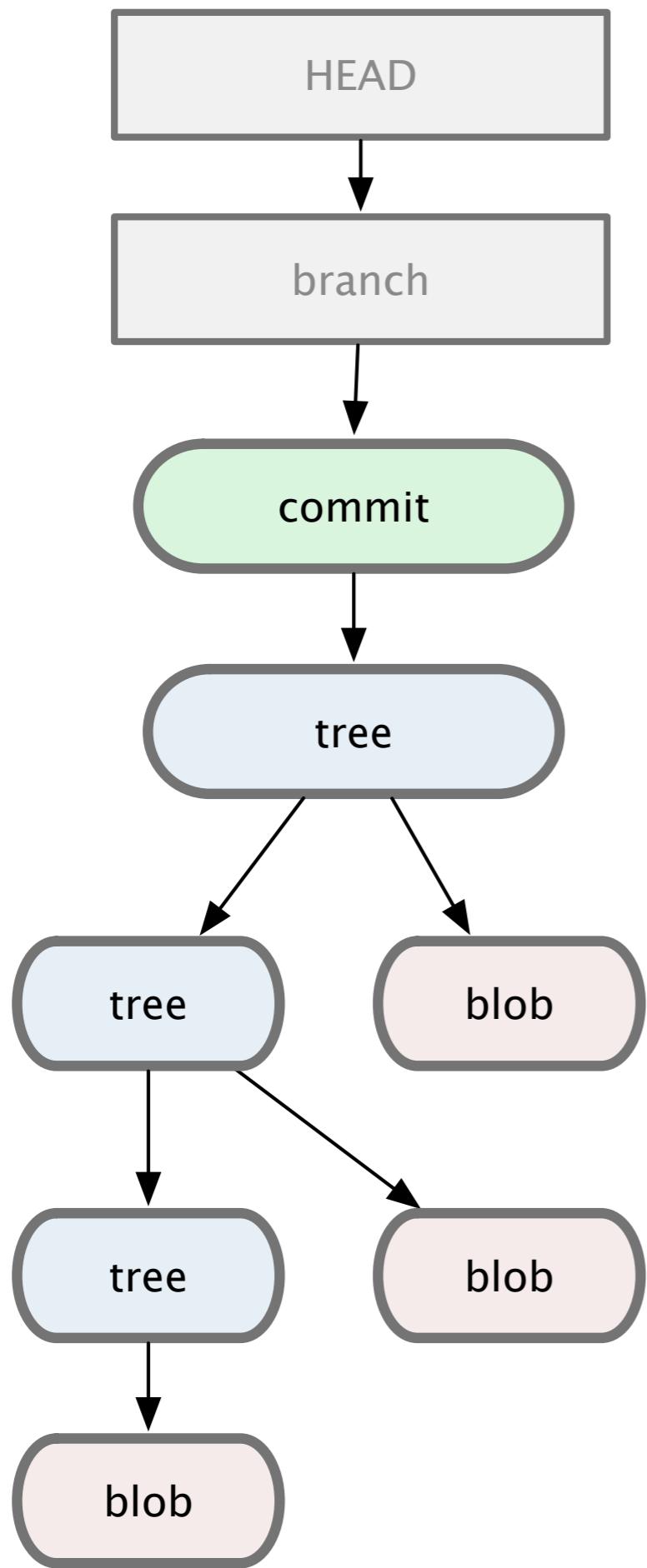
stored in `.git/refs/*`
as simple files

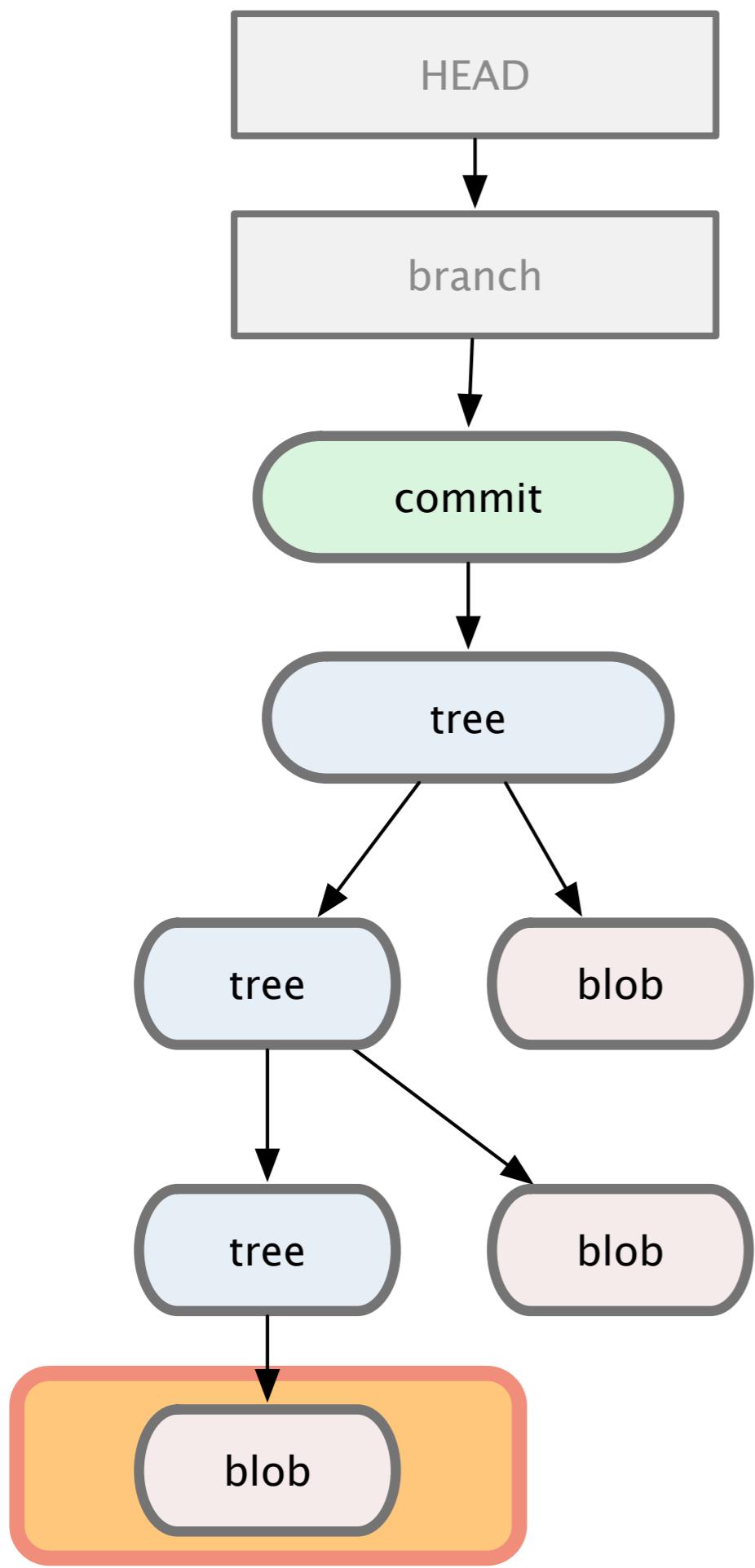


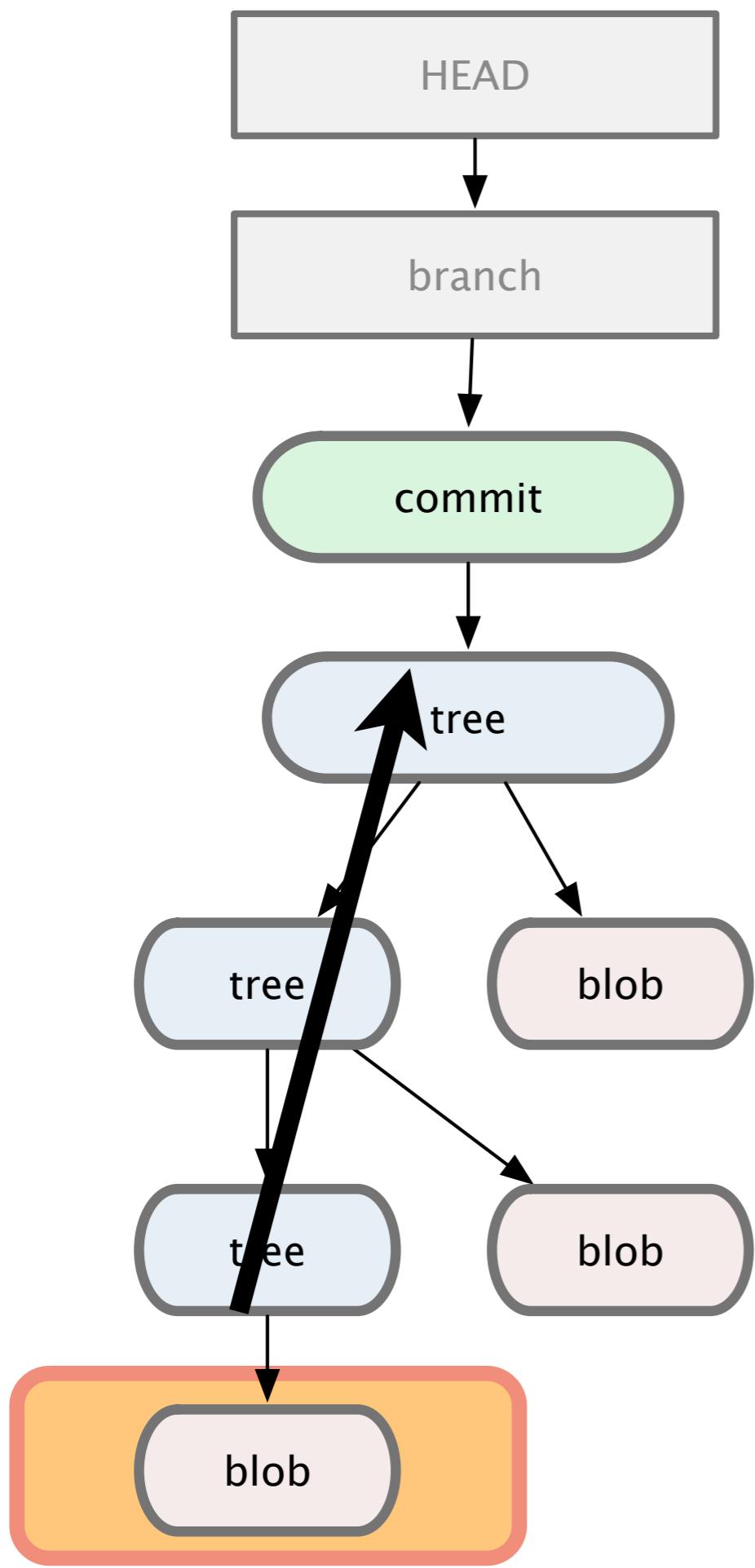
references

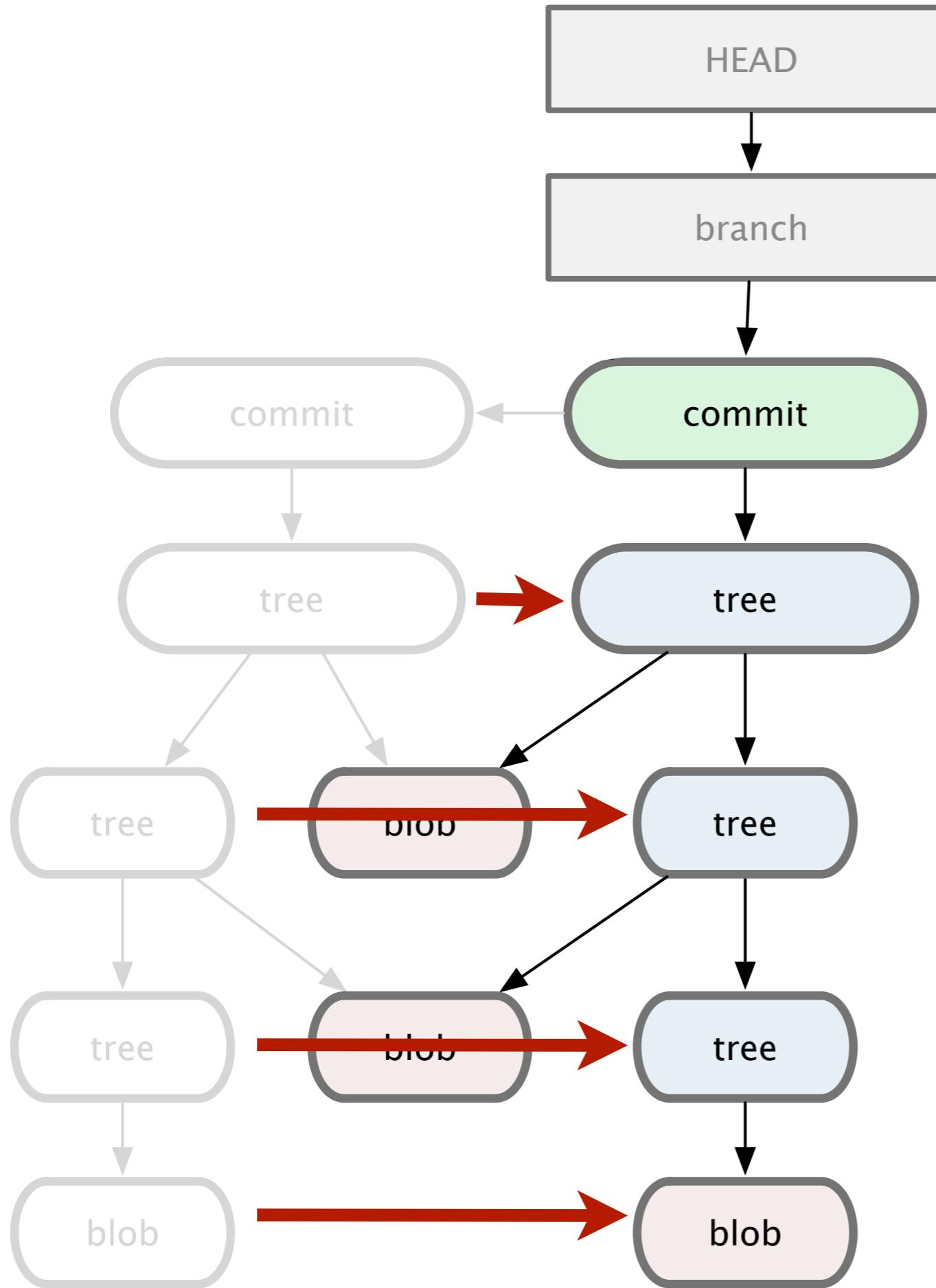


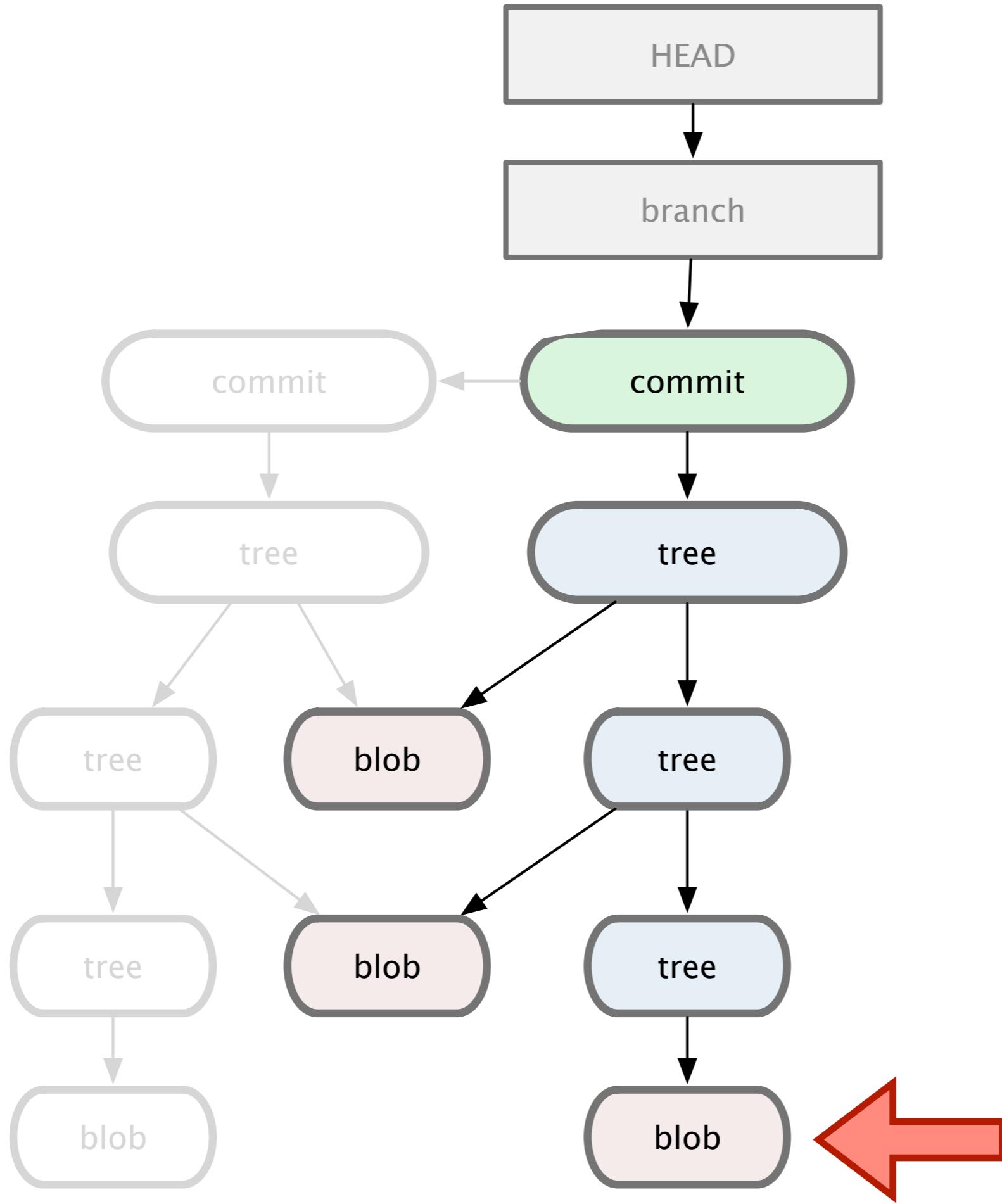
the object model

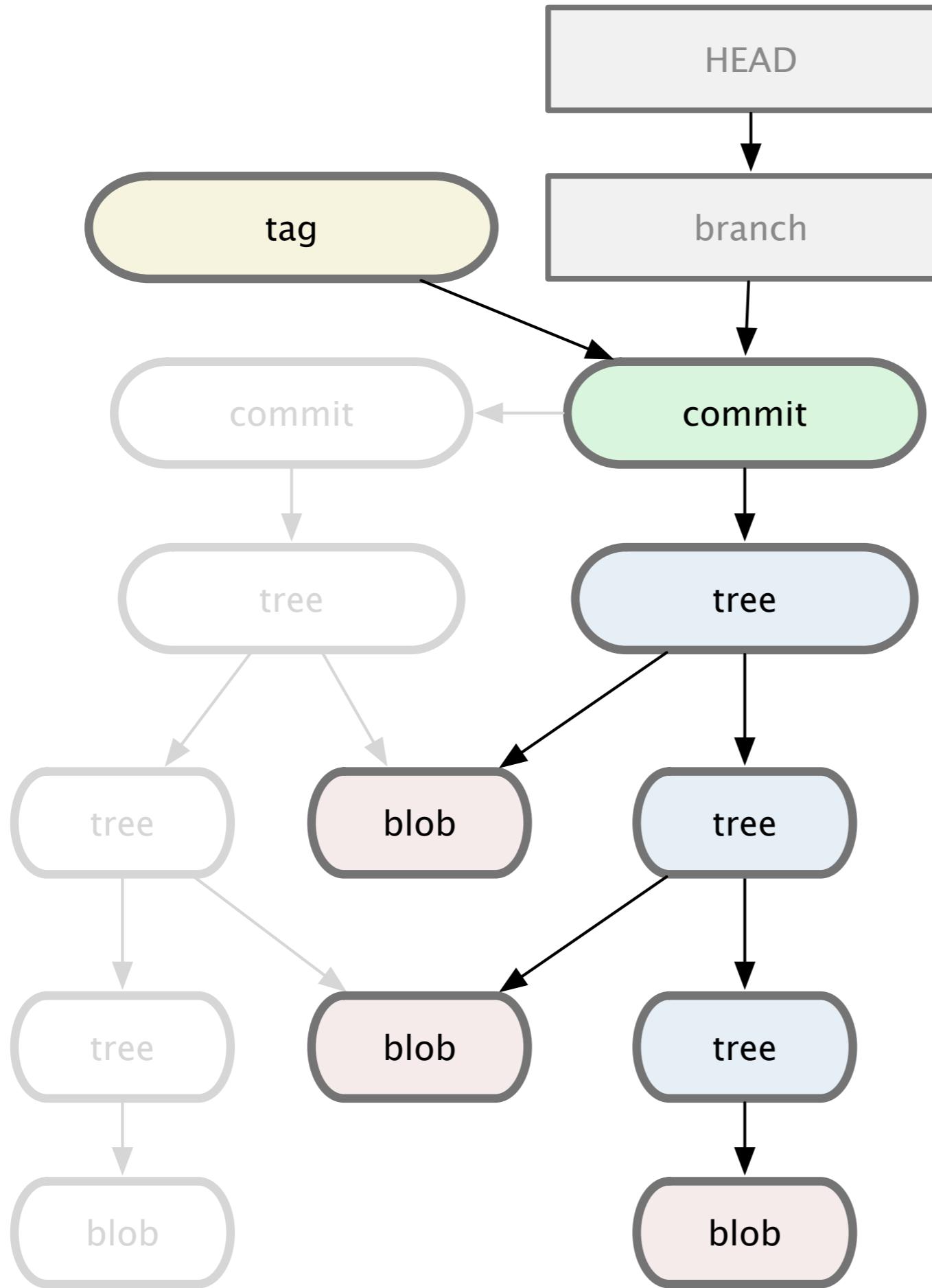


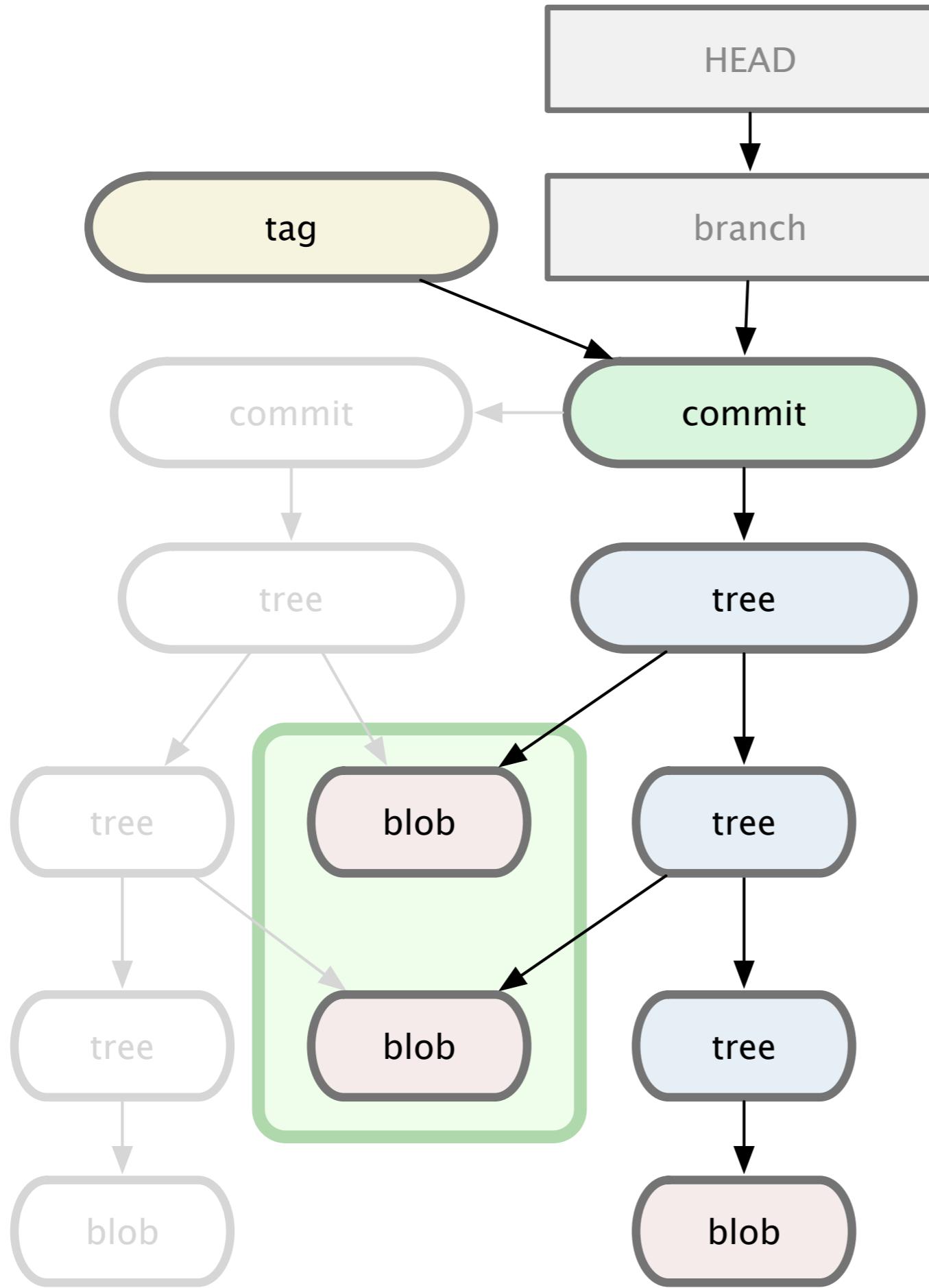


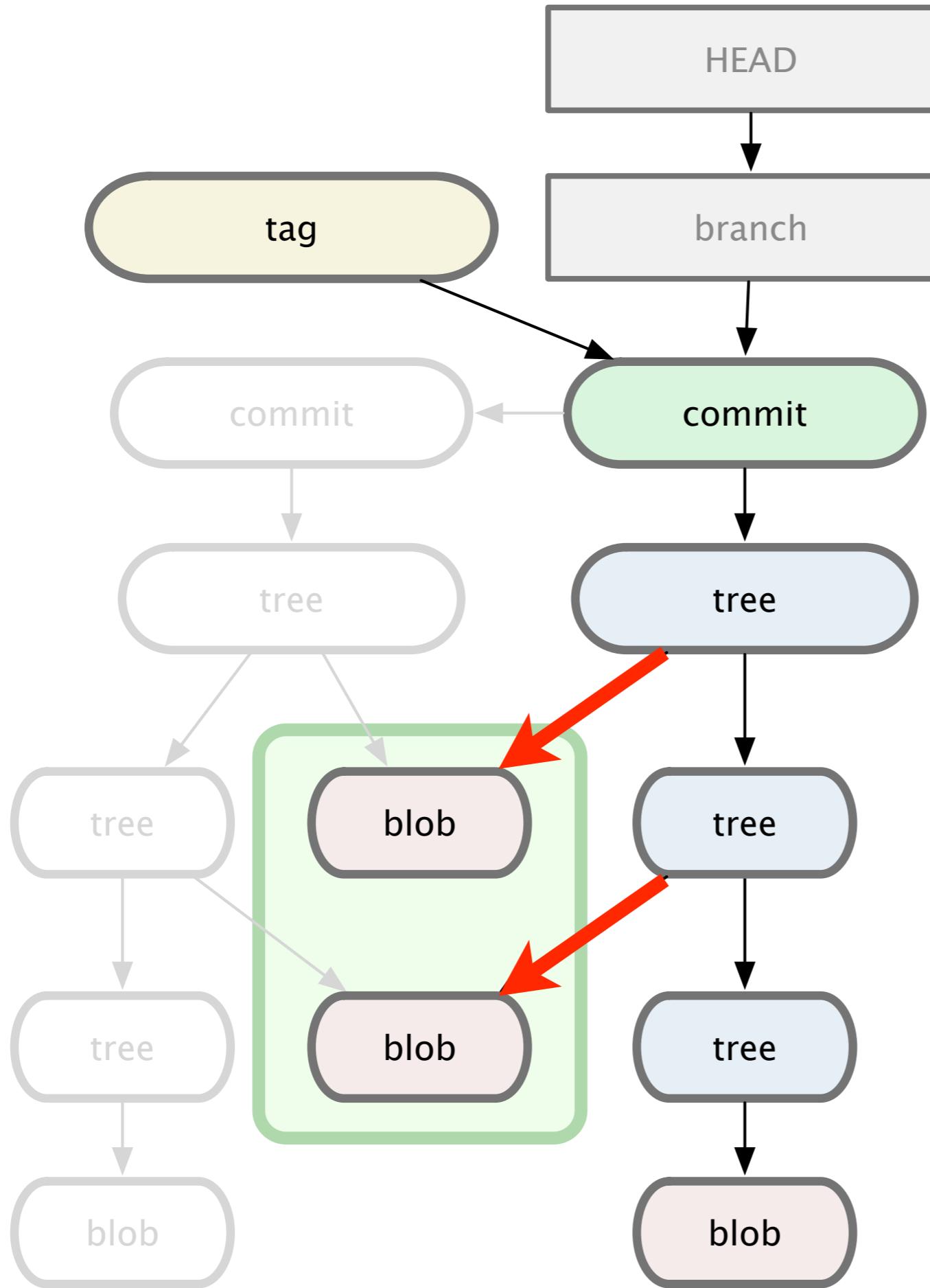


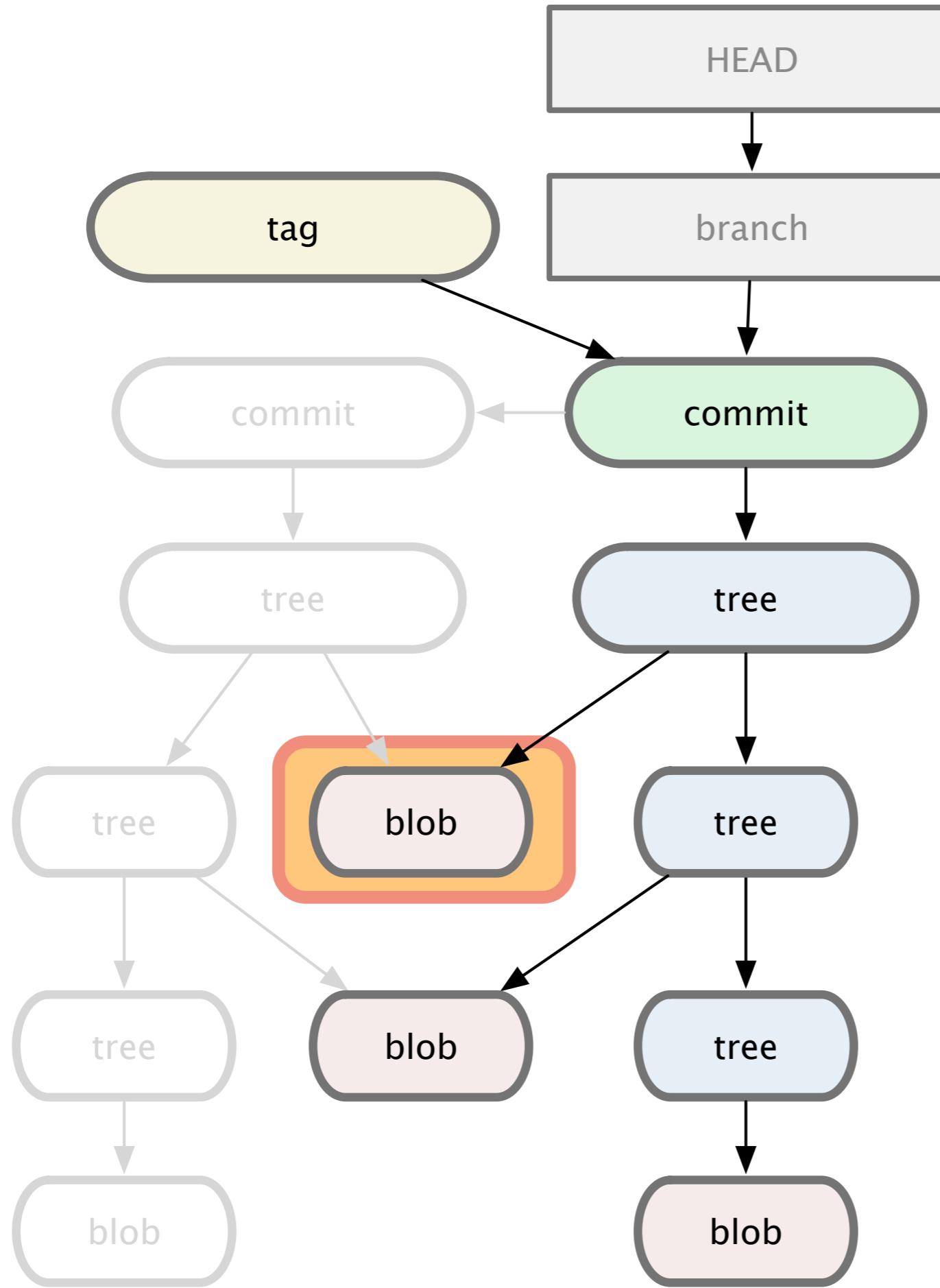


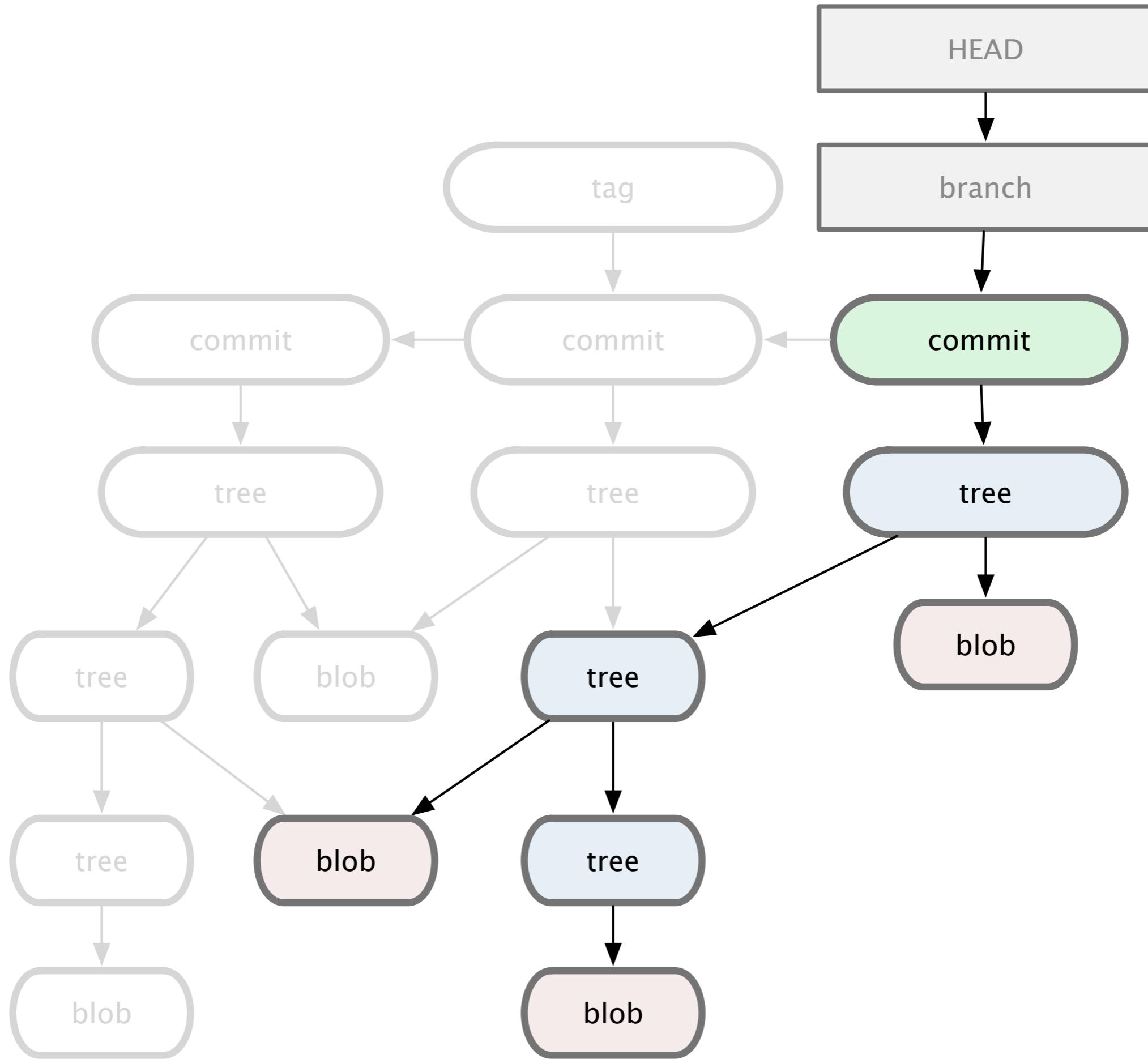


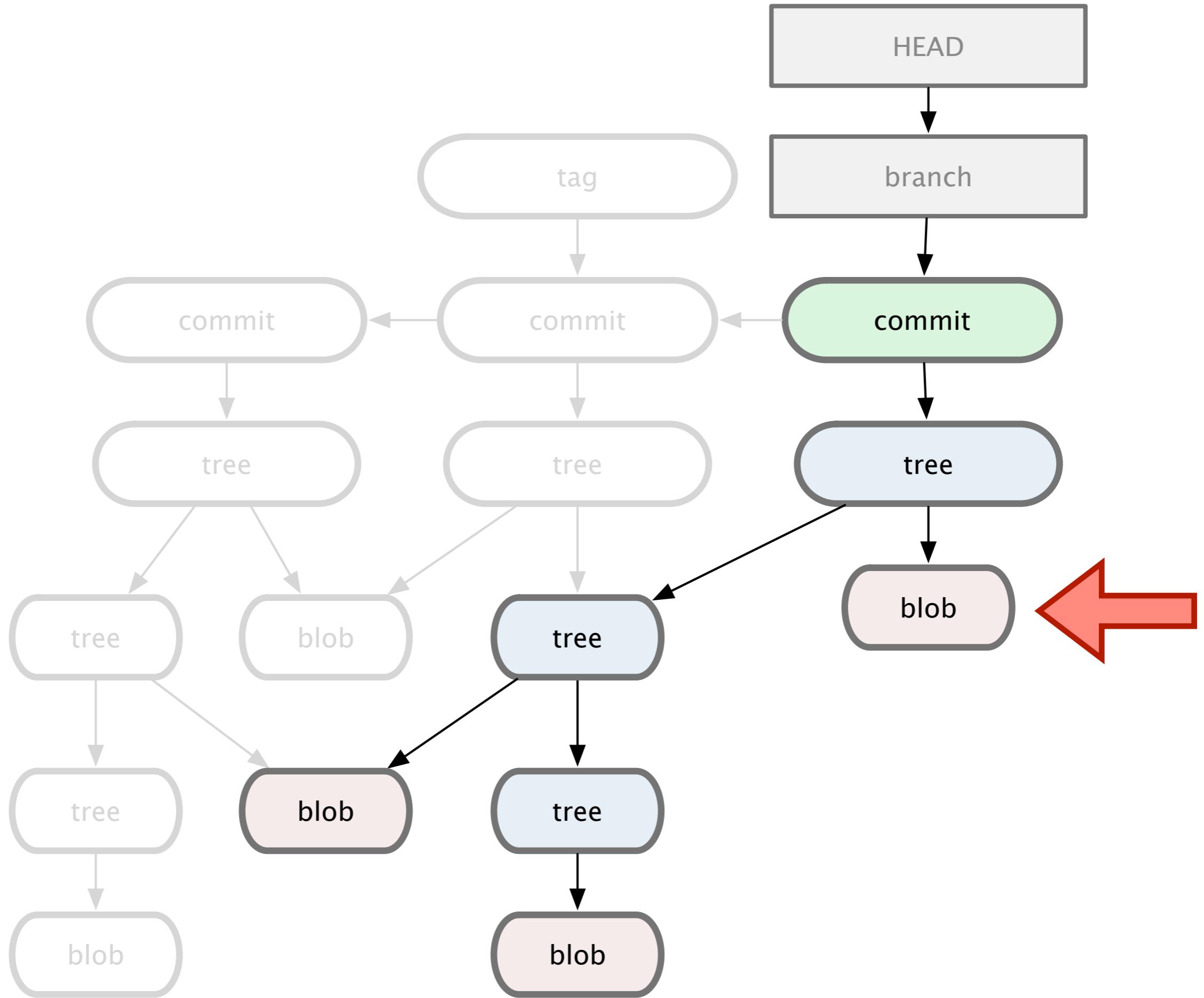


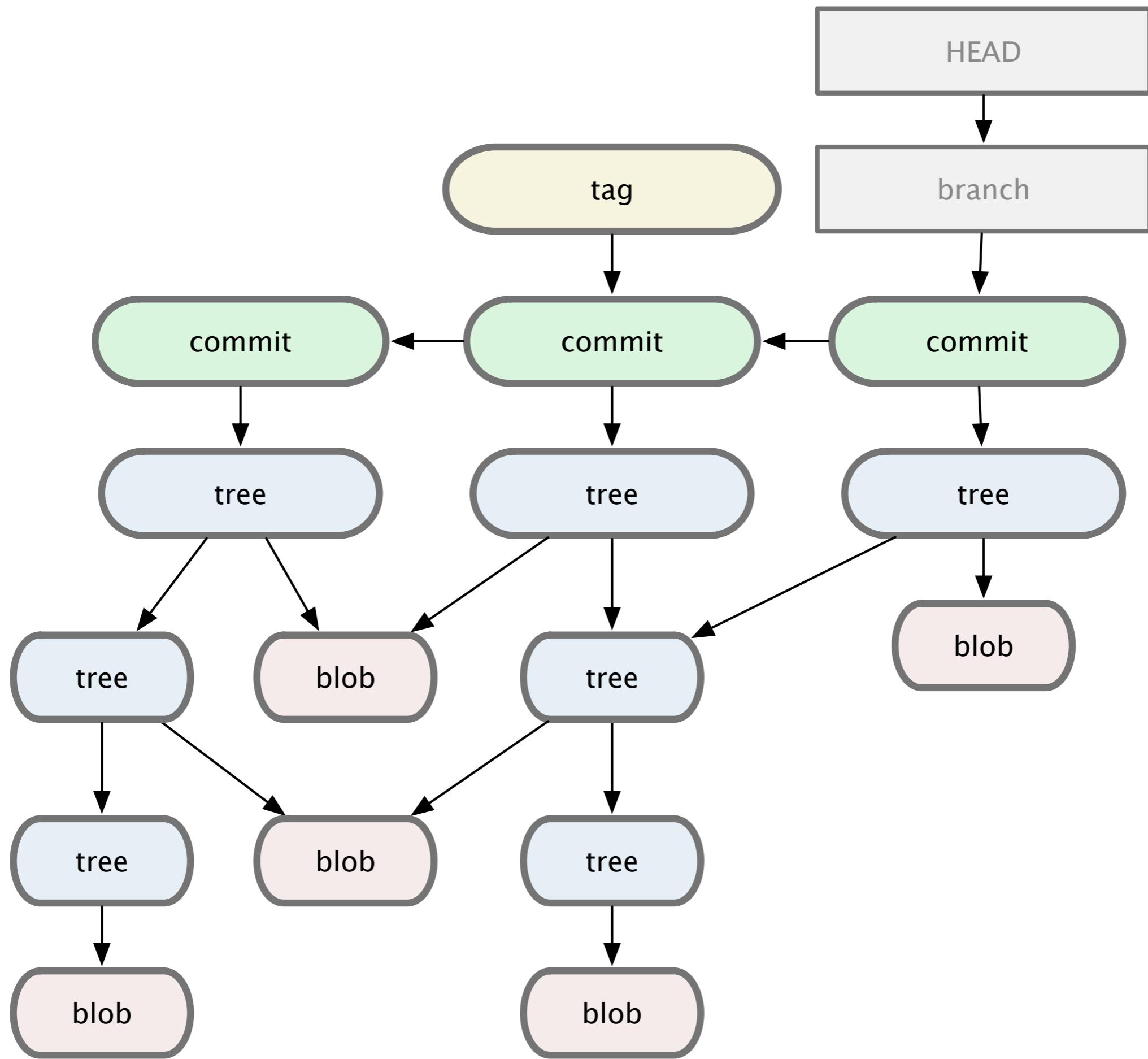


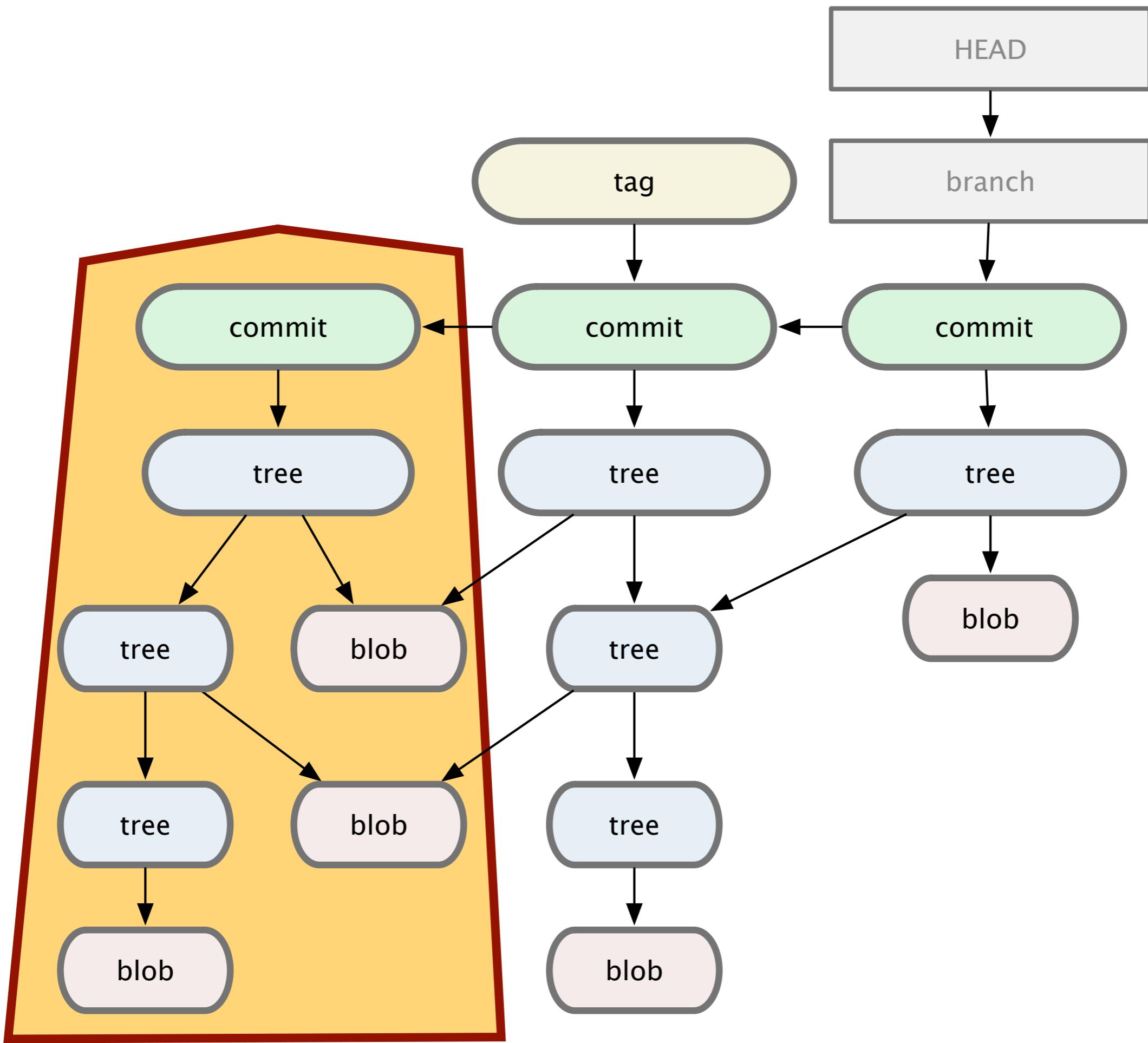


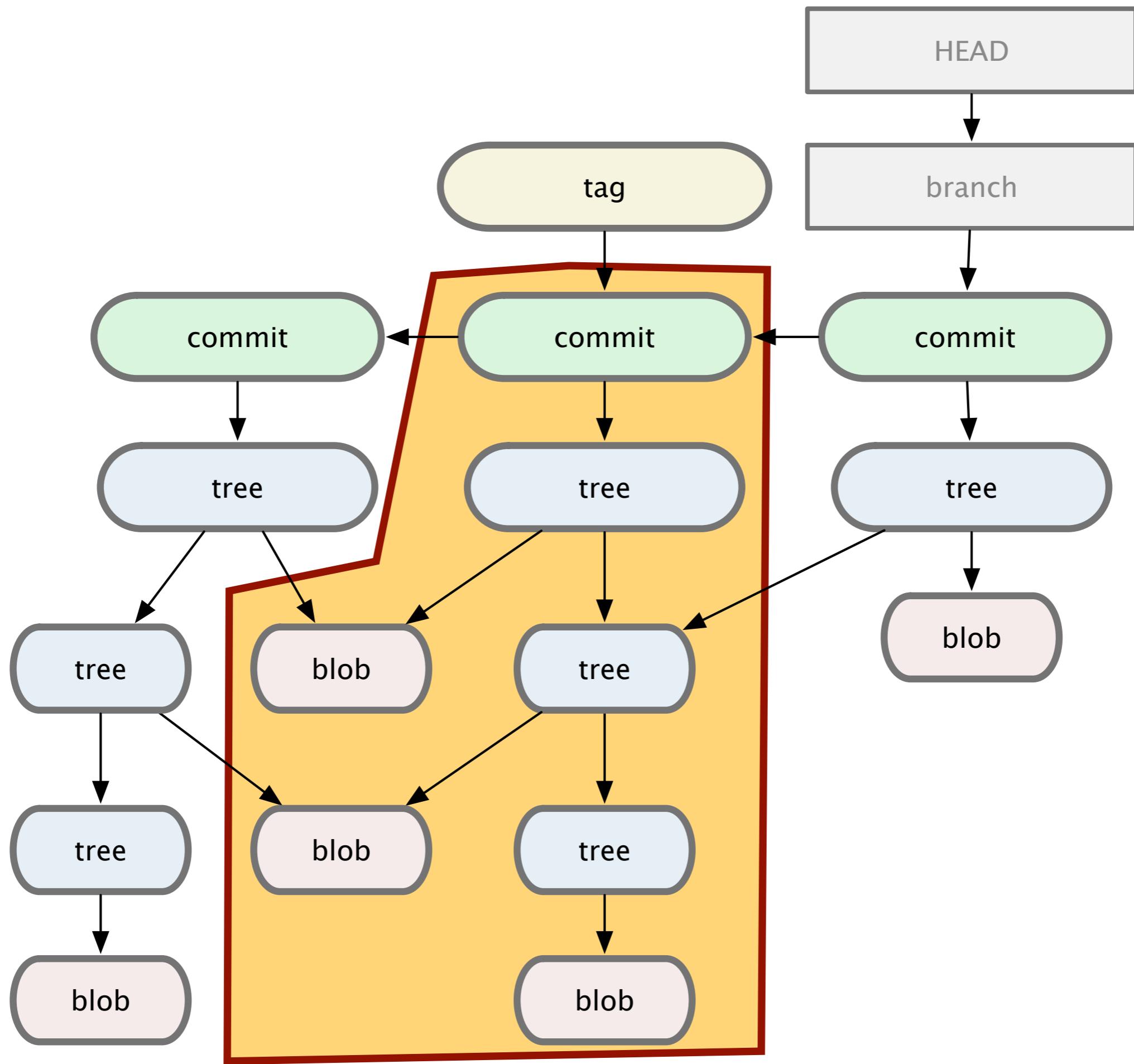


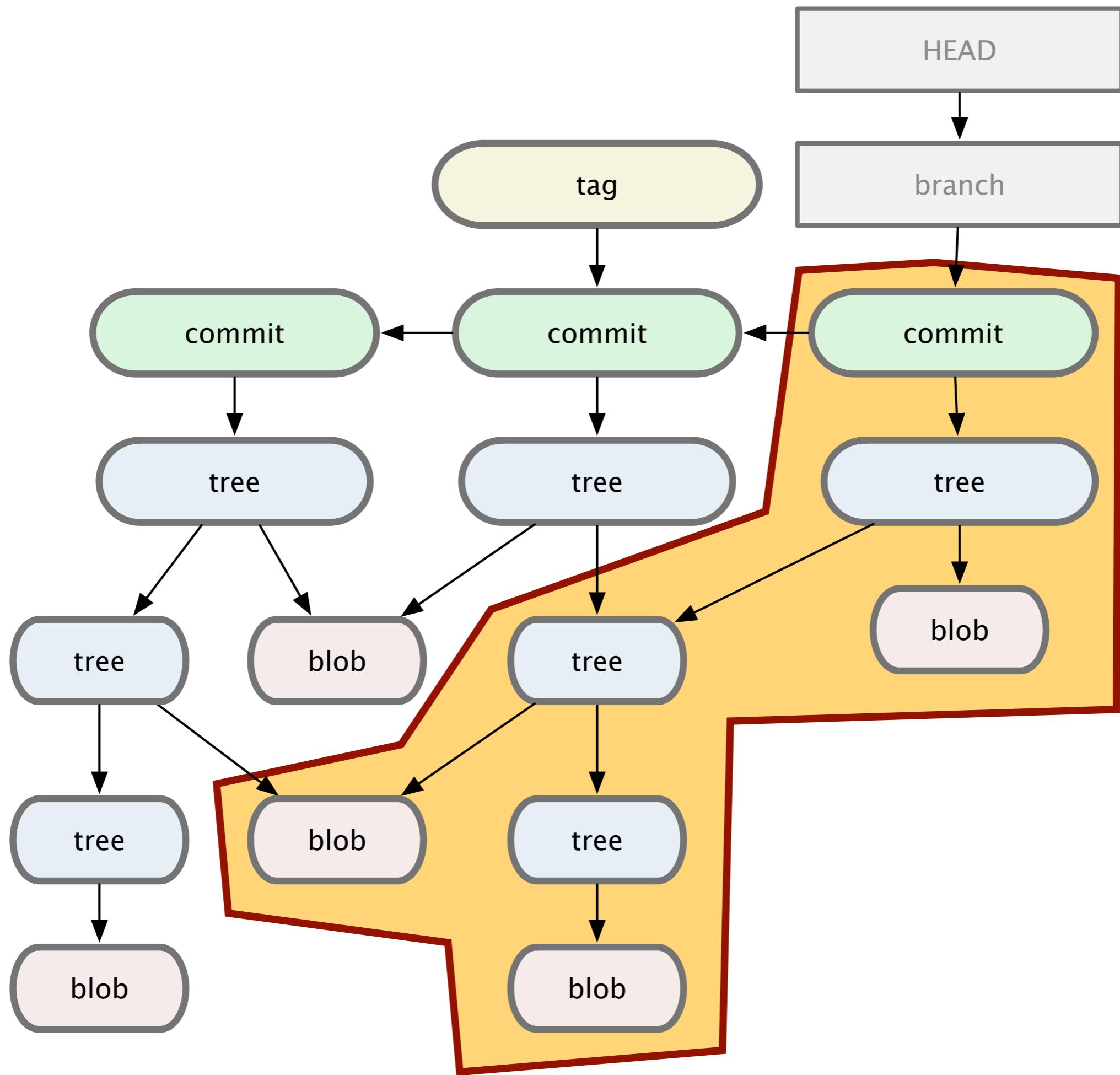












1

```
$ git checkout v0.1
```

1

```
$ git checkout v0.1
```



2

```
.git/refs/tags/v0.1
```

1

```
$ git checkout v0.1
```

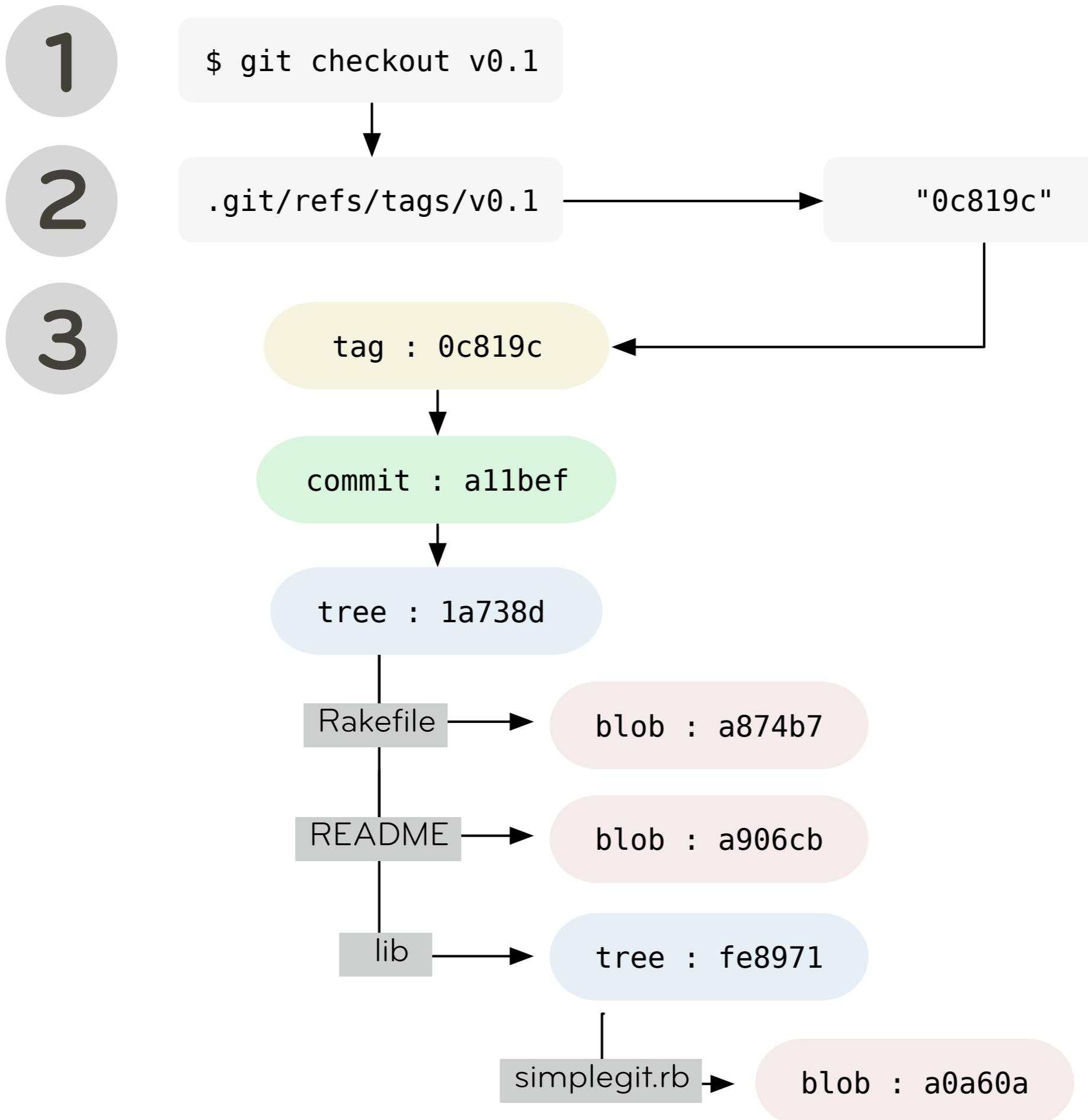


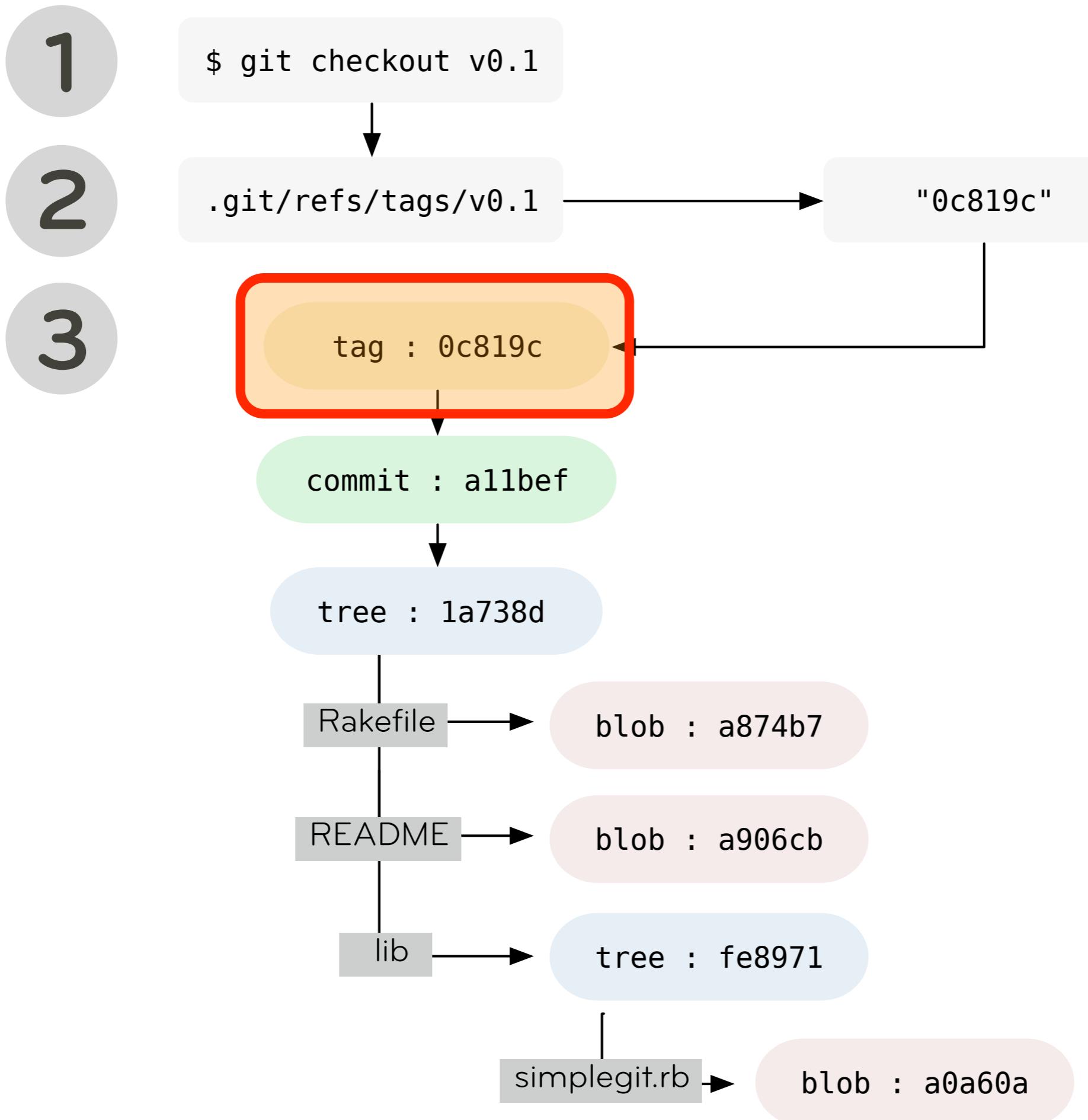
2

```
.git/refs/tags/v0.1
```

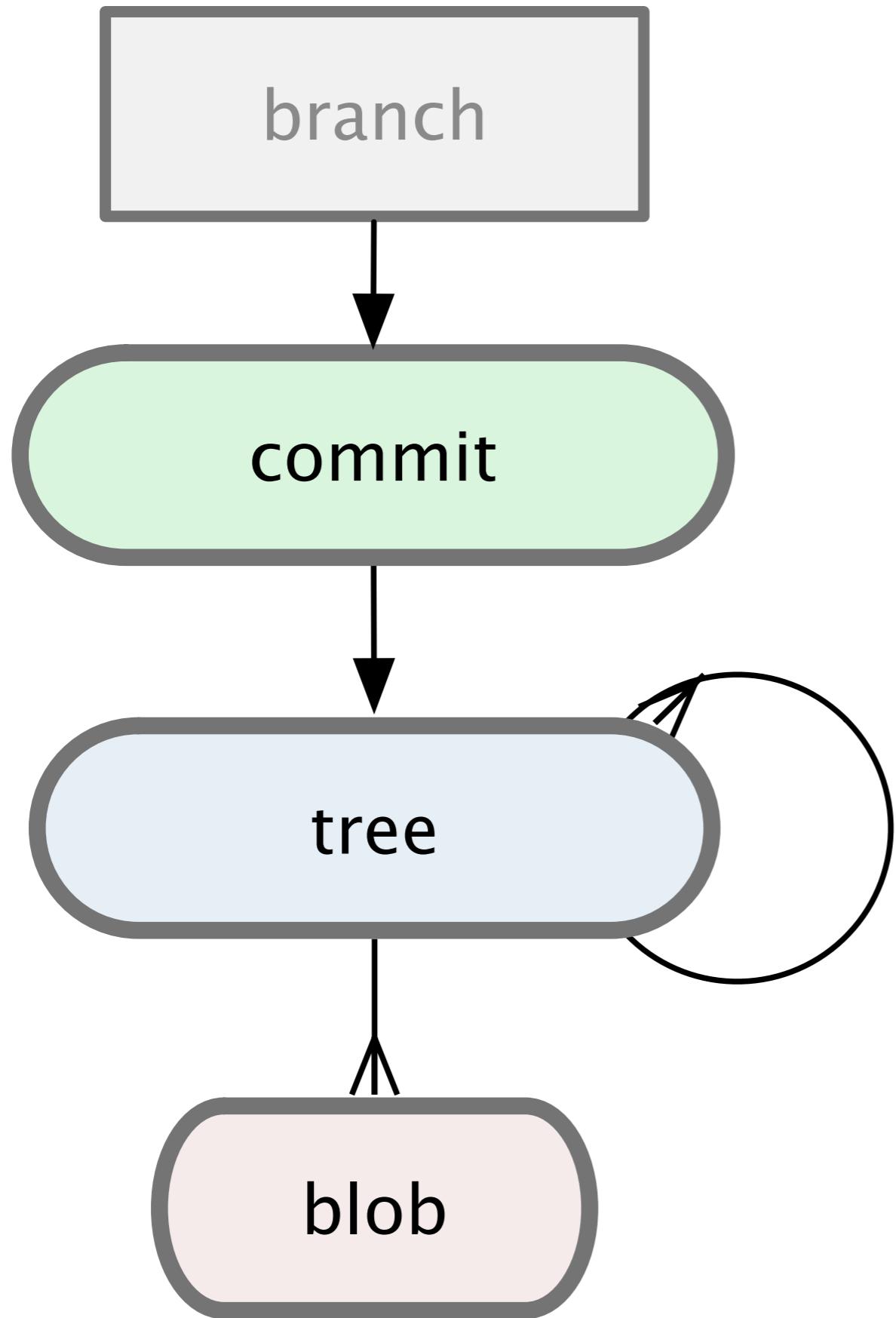


```
"0c819c"
```

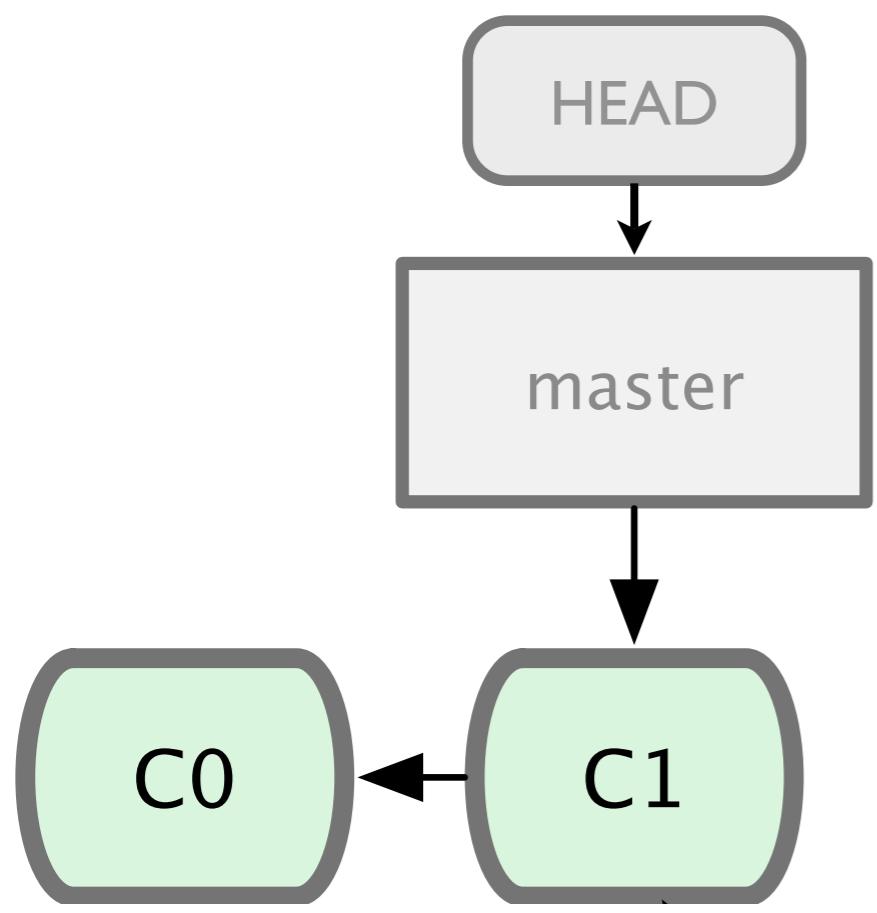


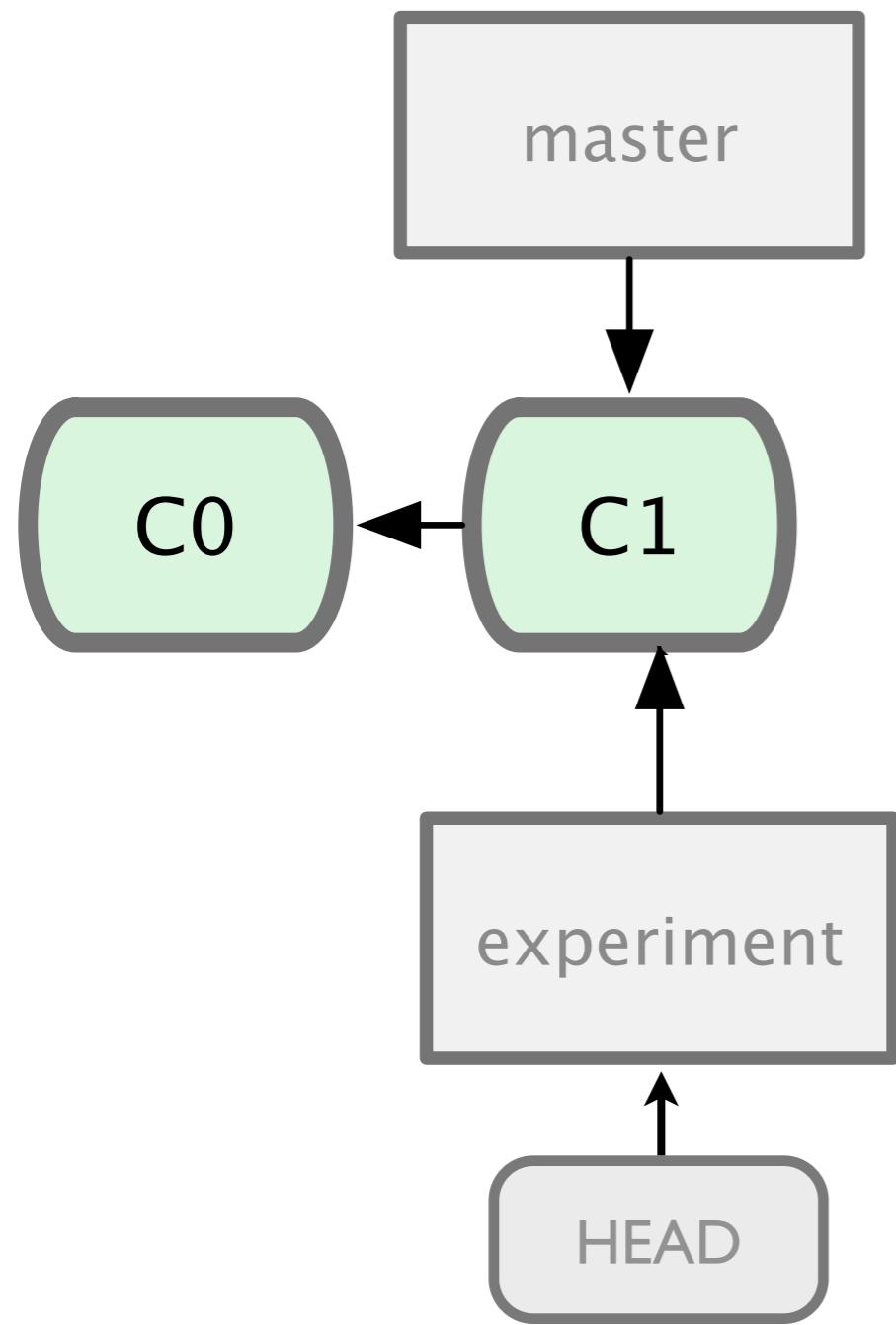


the commit history

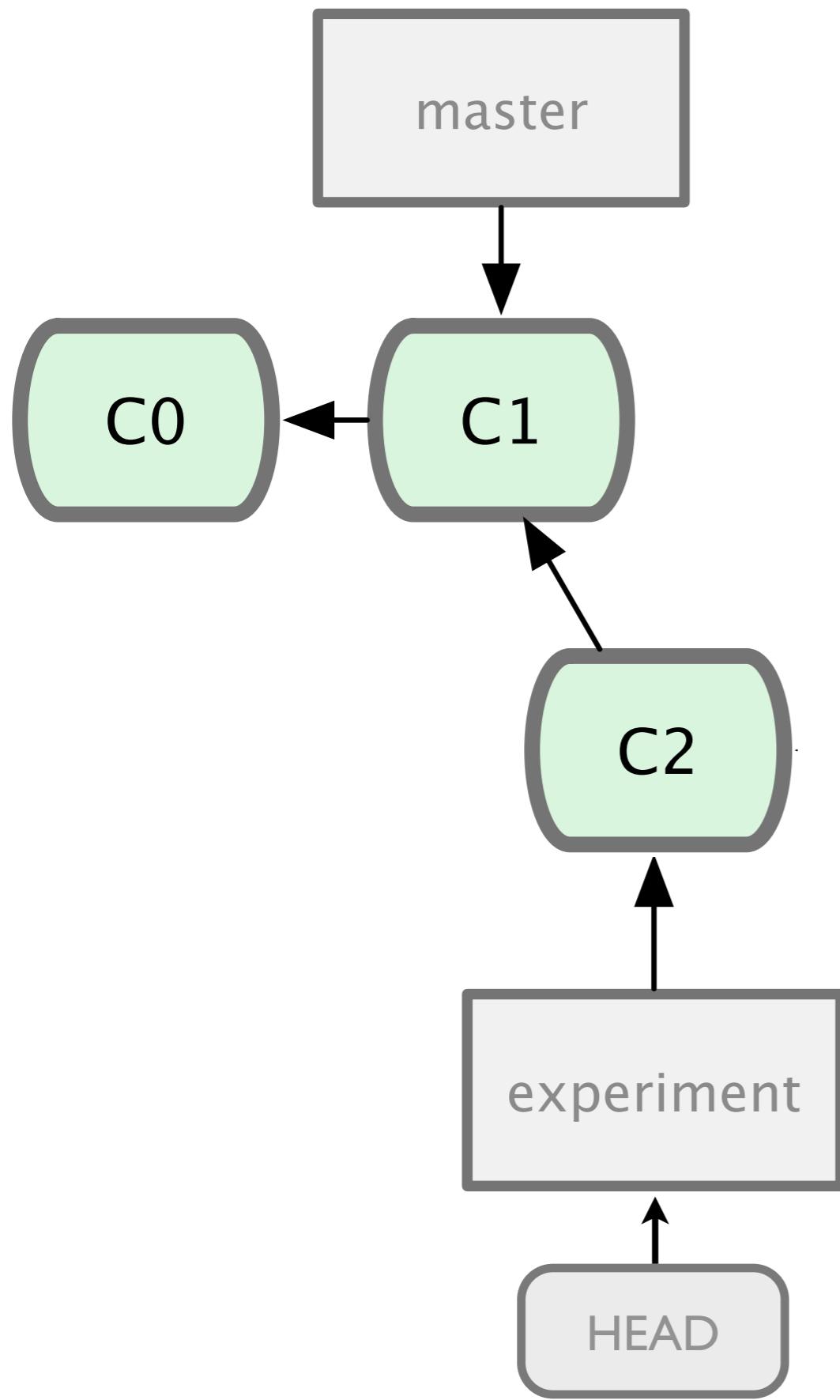


branching and merging

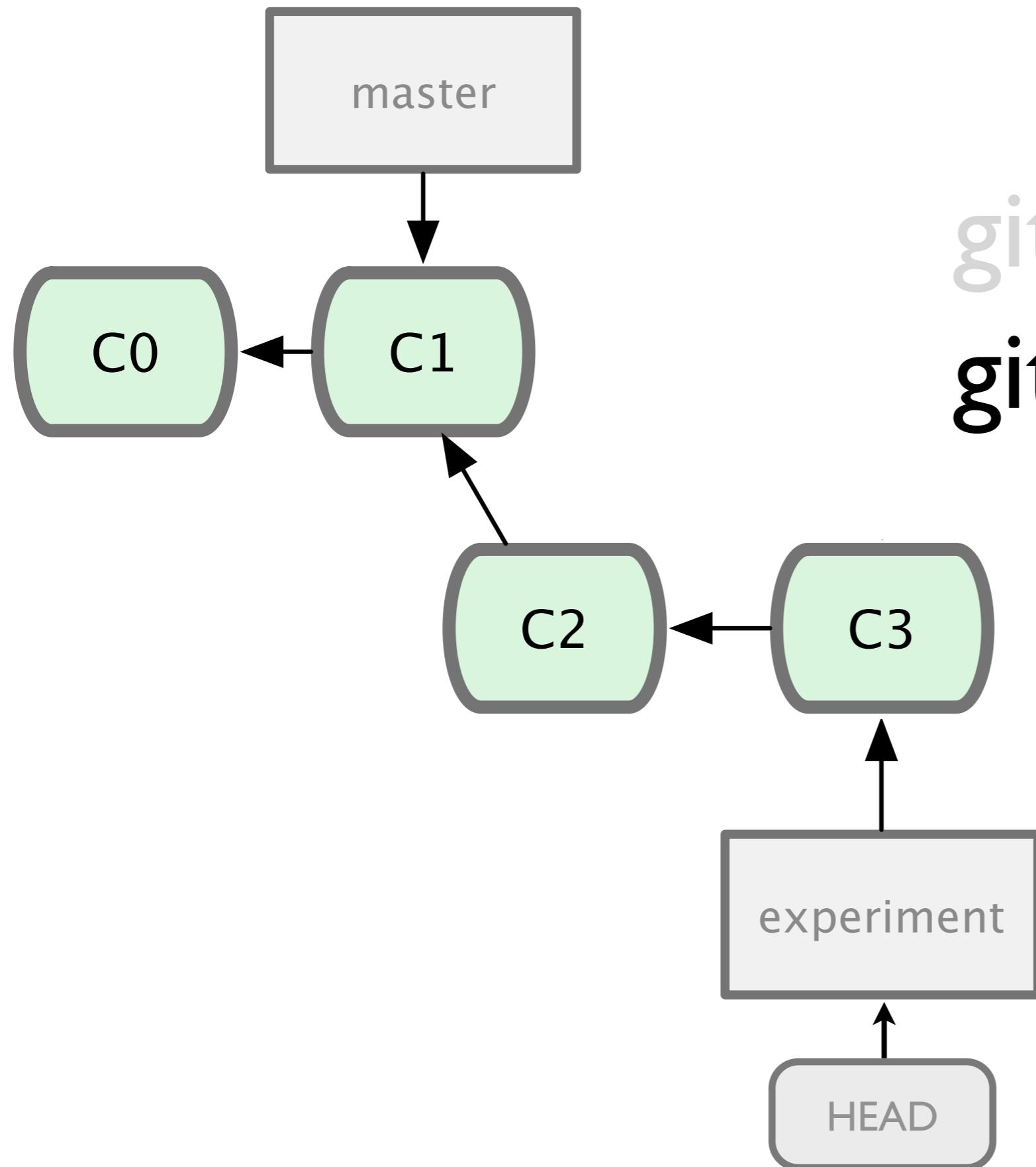




**git checkout -b
experiment**

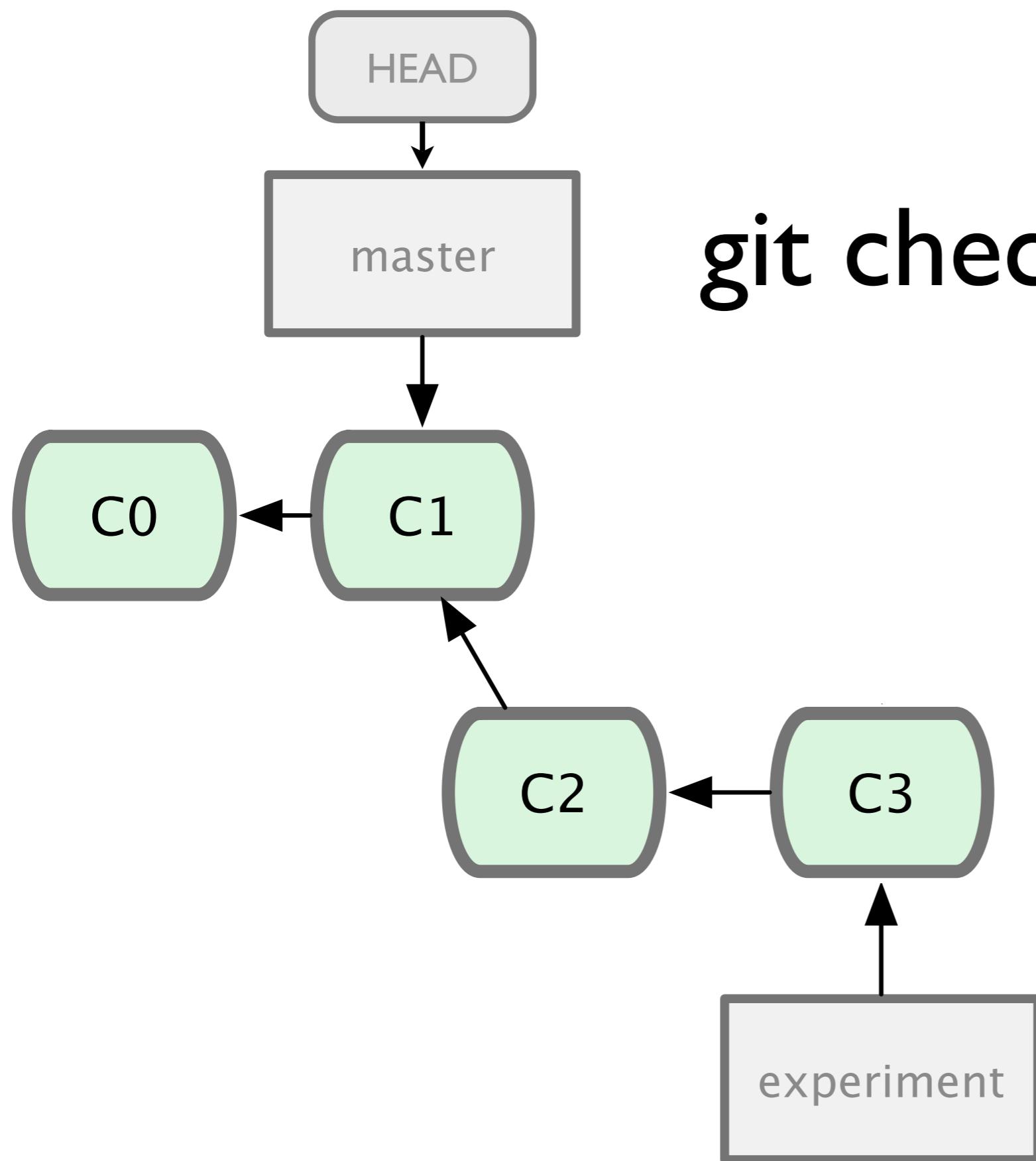


git commit

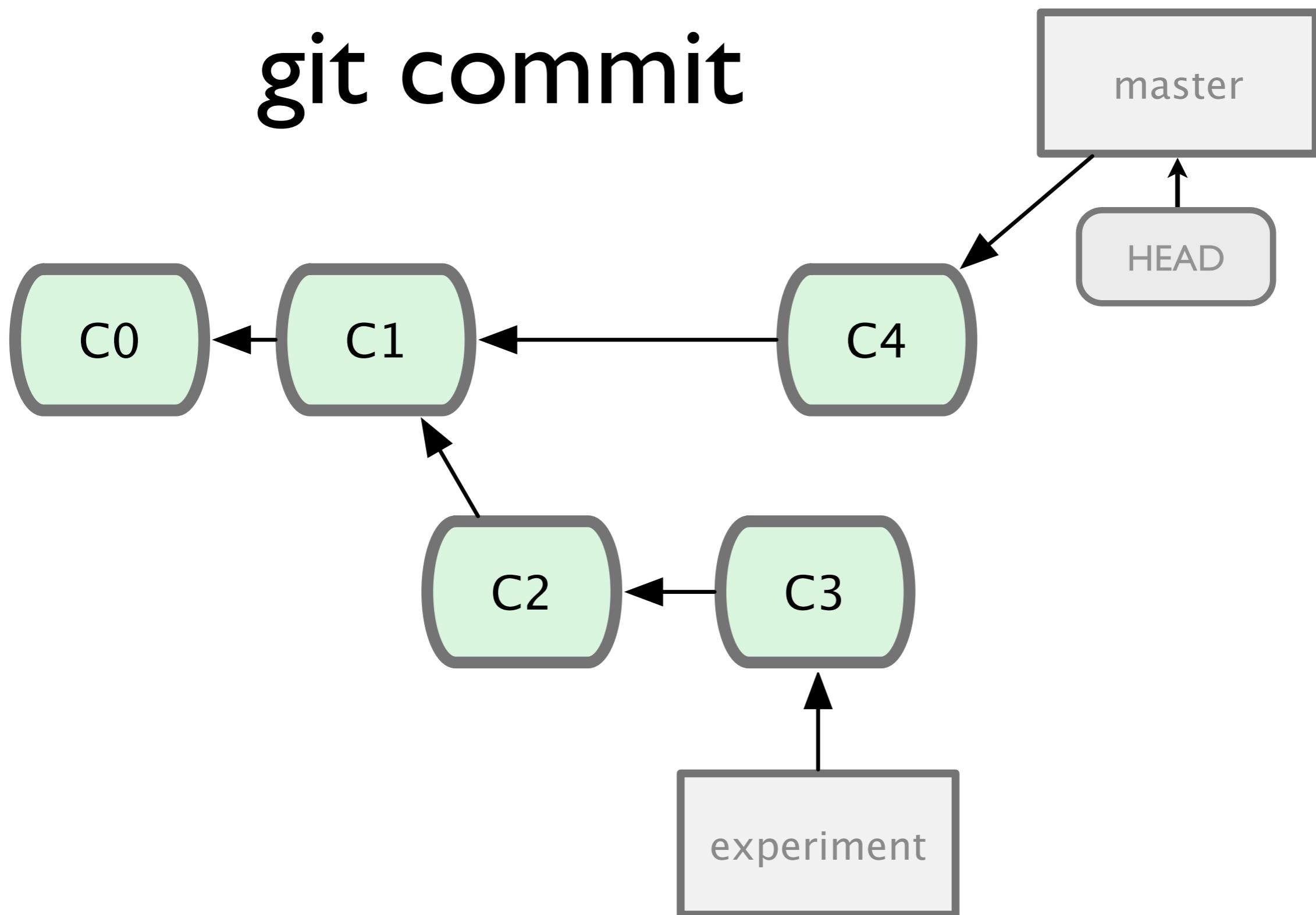


git commit
git commit

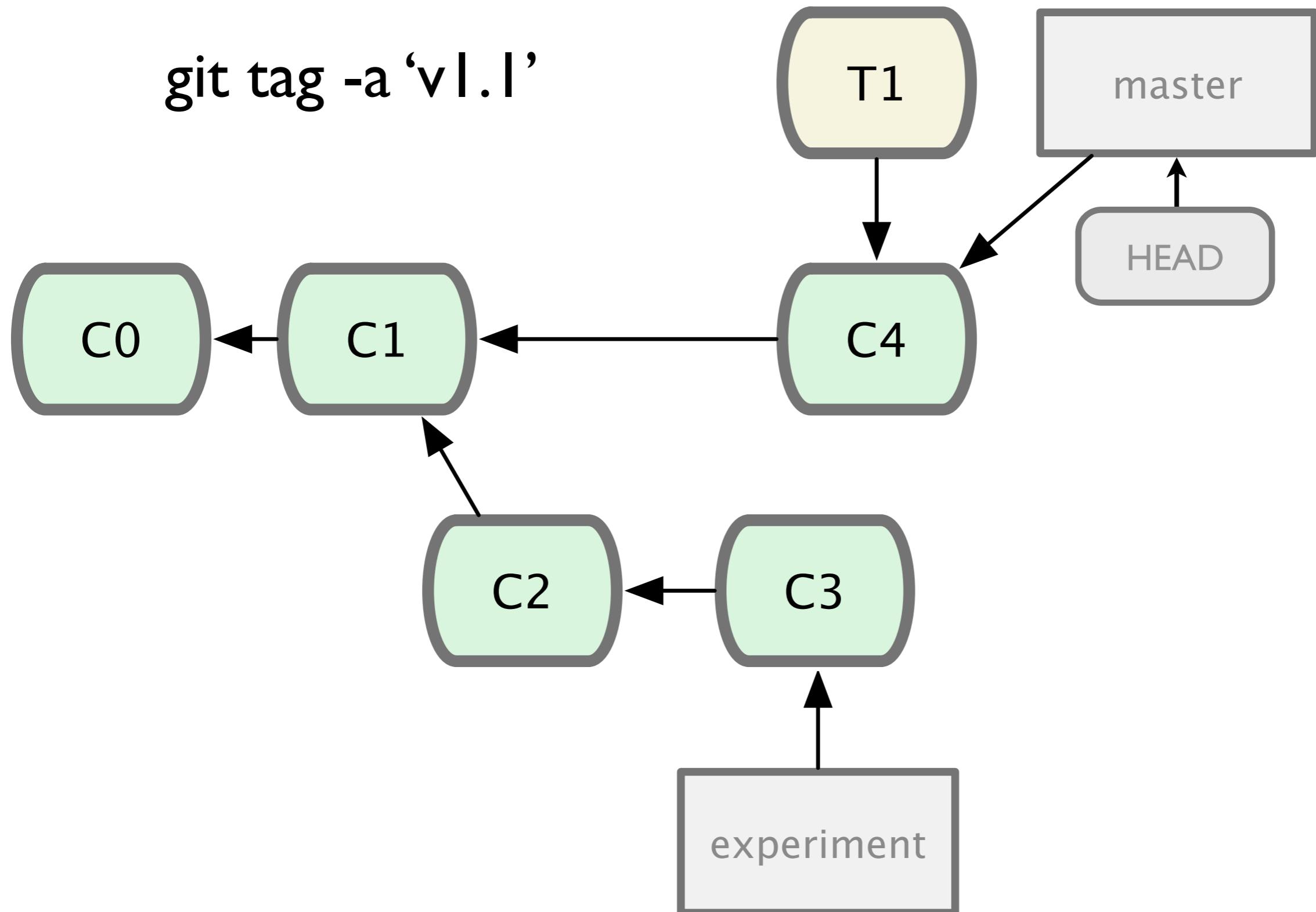
git checkout master

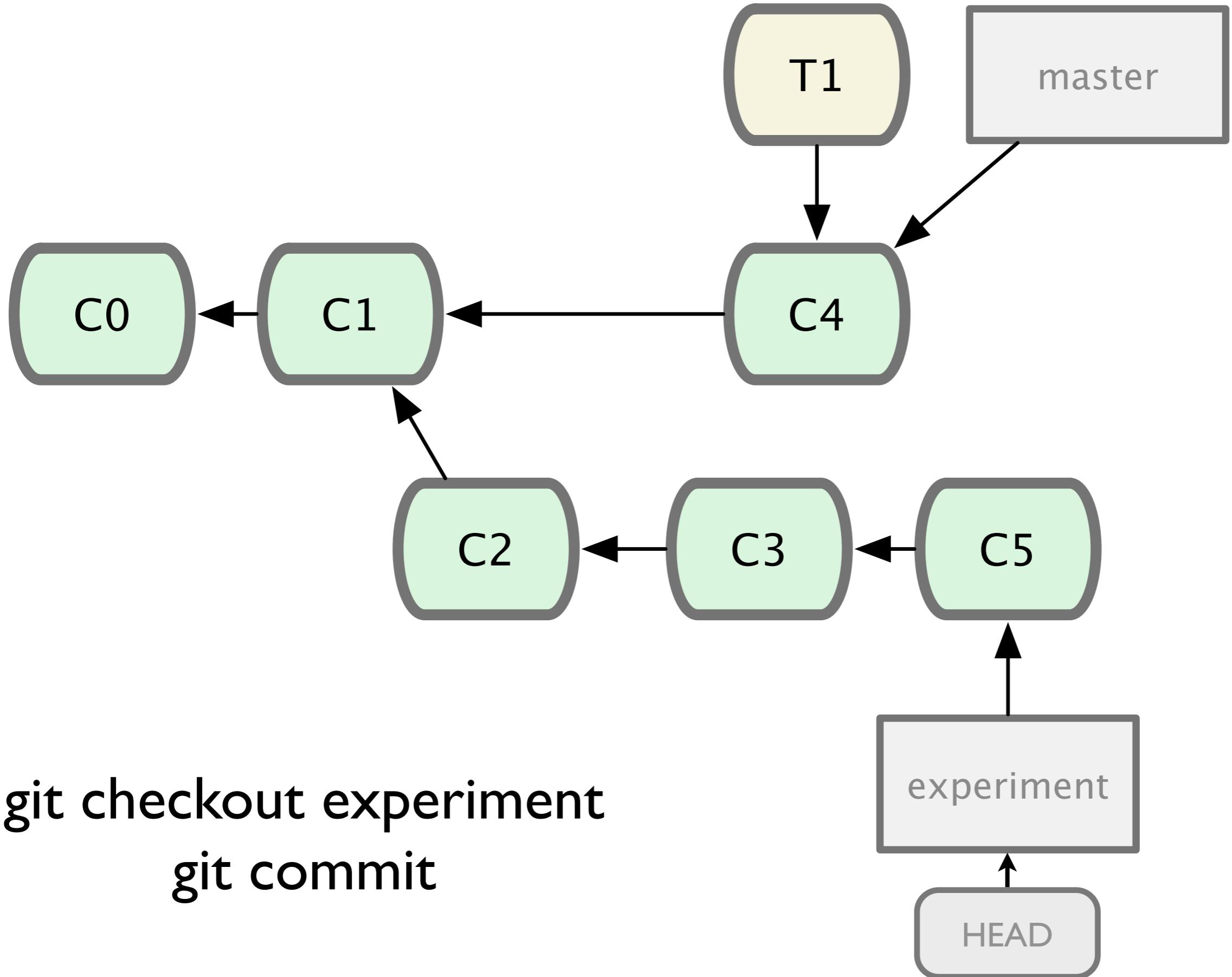


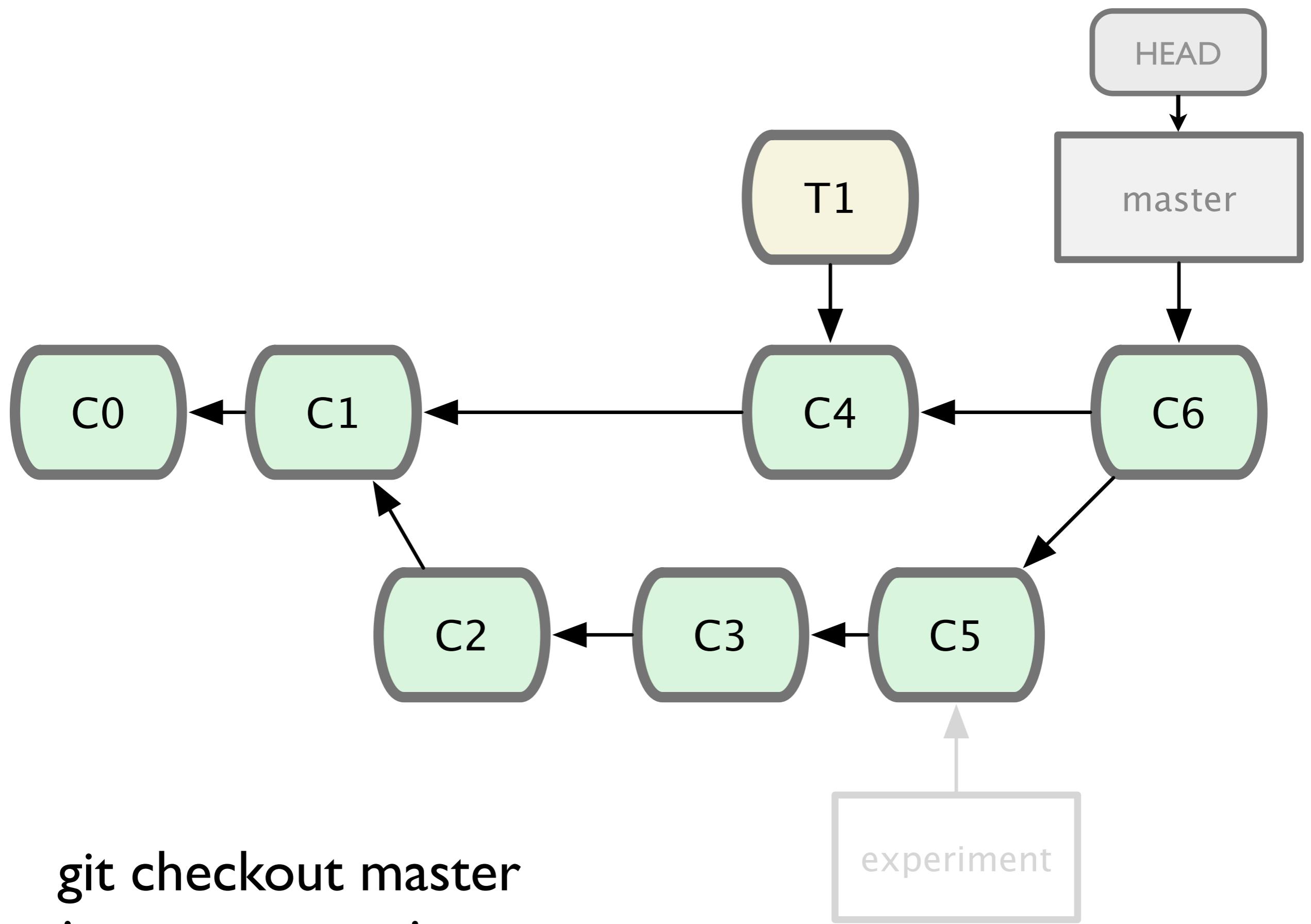
git commit



`git tag -a 'v1.1'`







git checkout master
git merge experiment

remotes

master



origin/master



C0

master

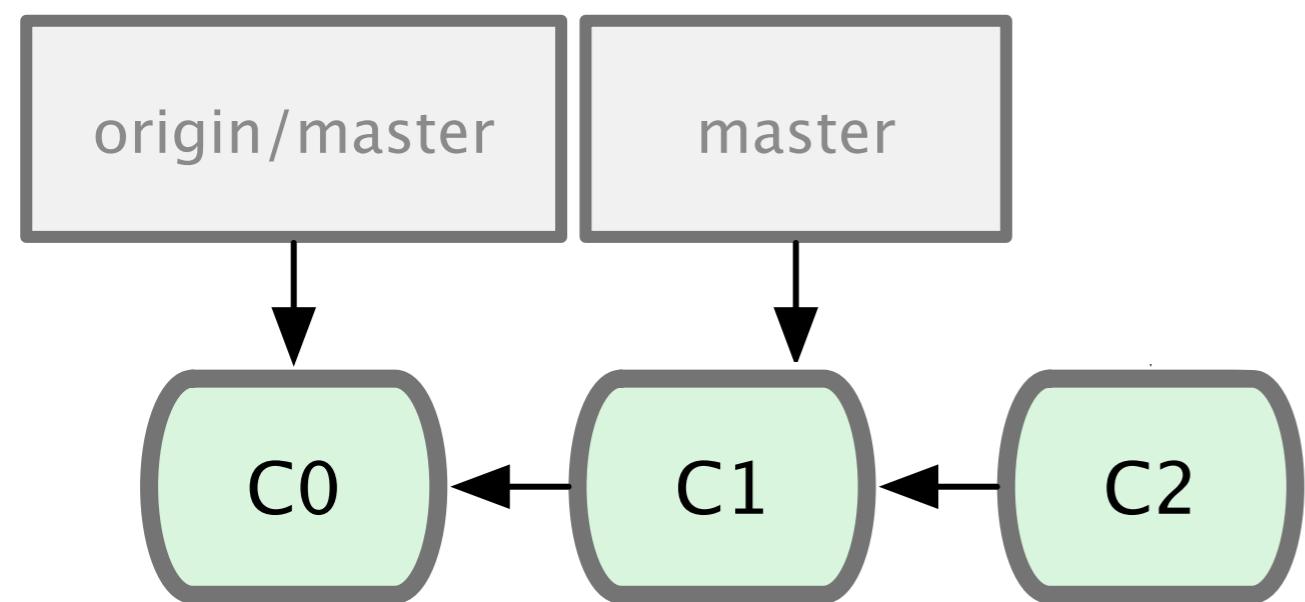


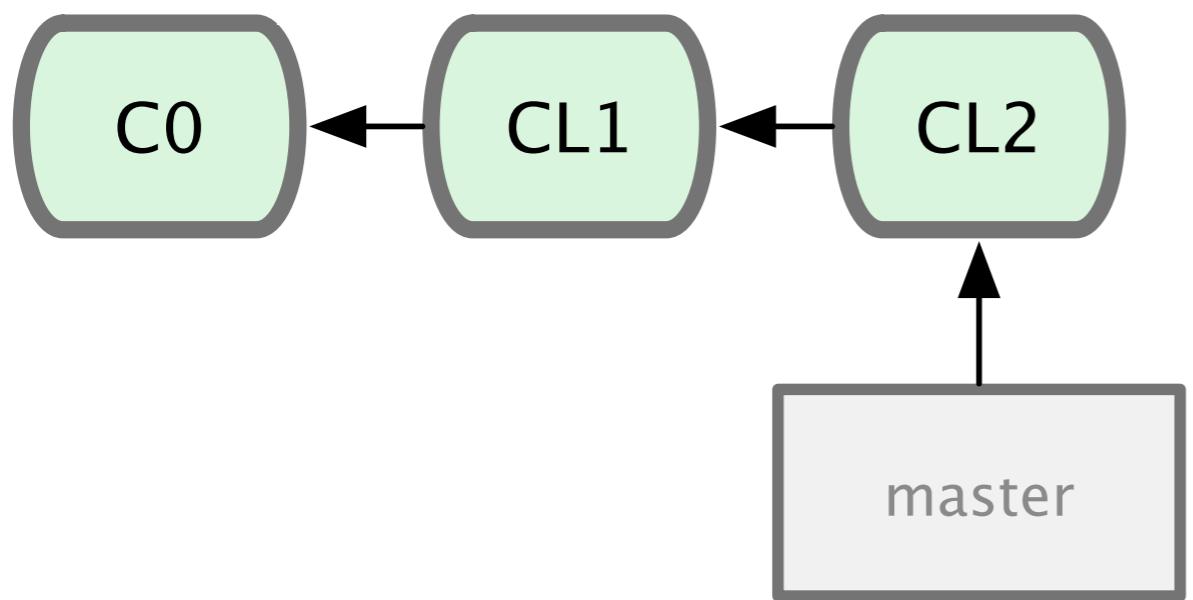
origin/master

C0

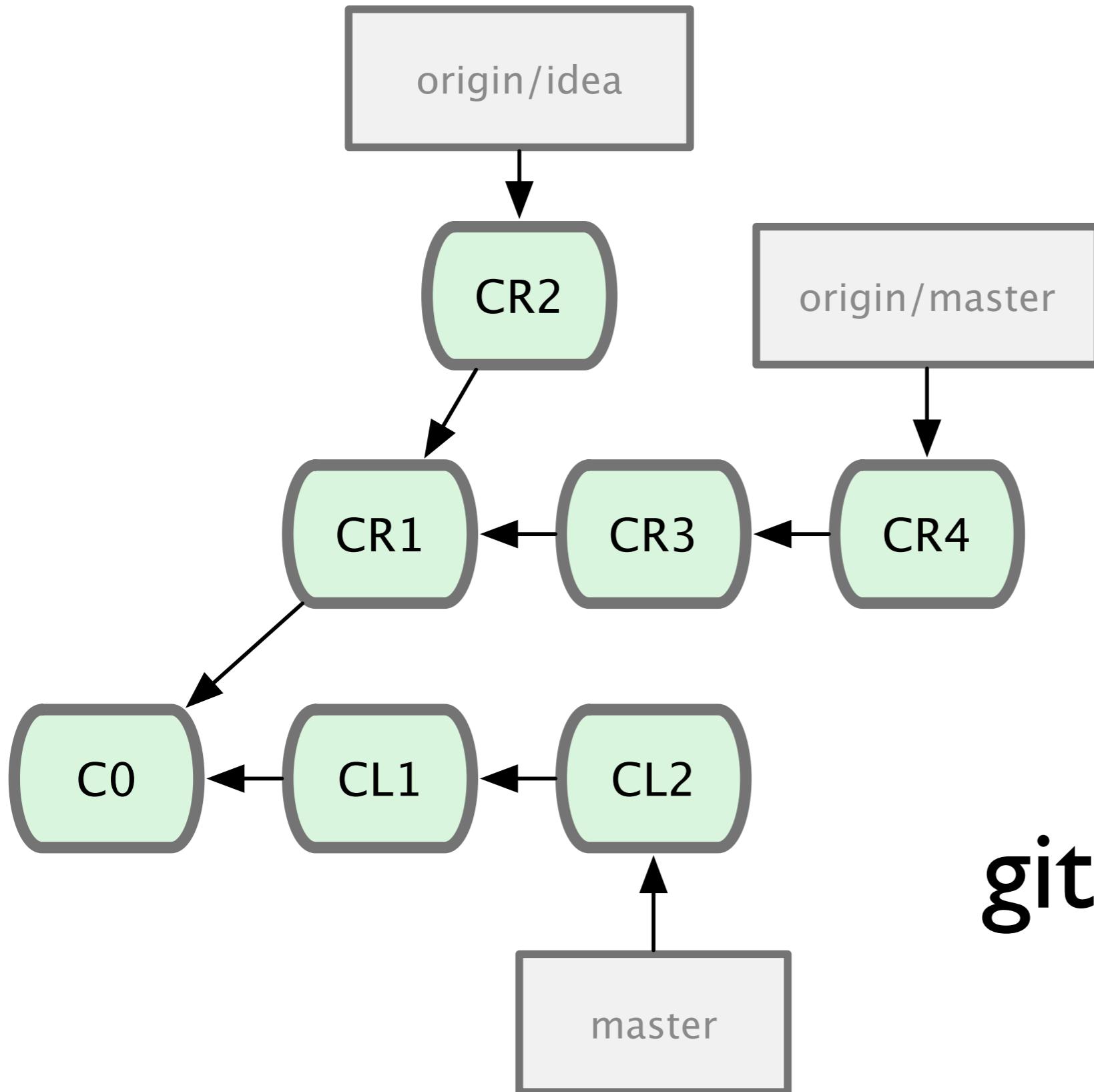
C1



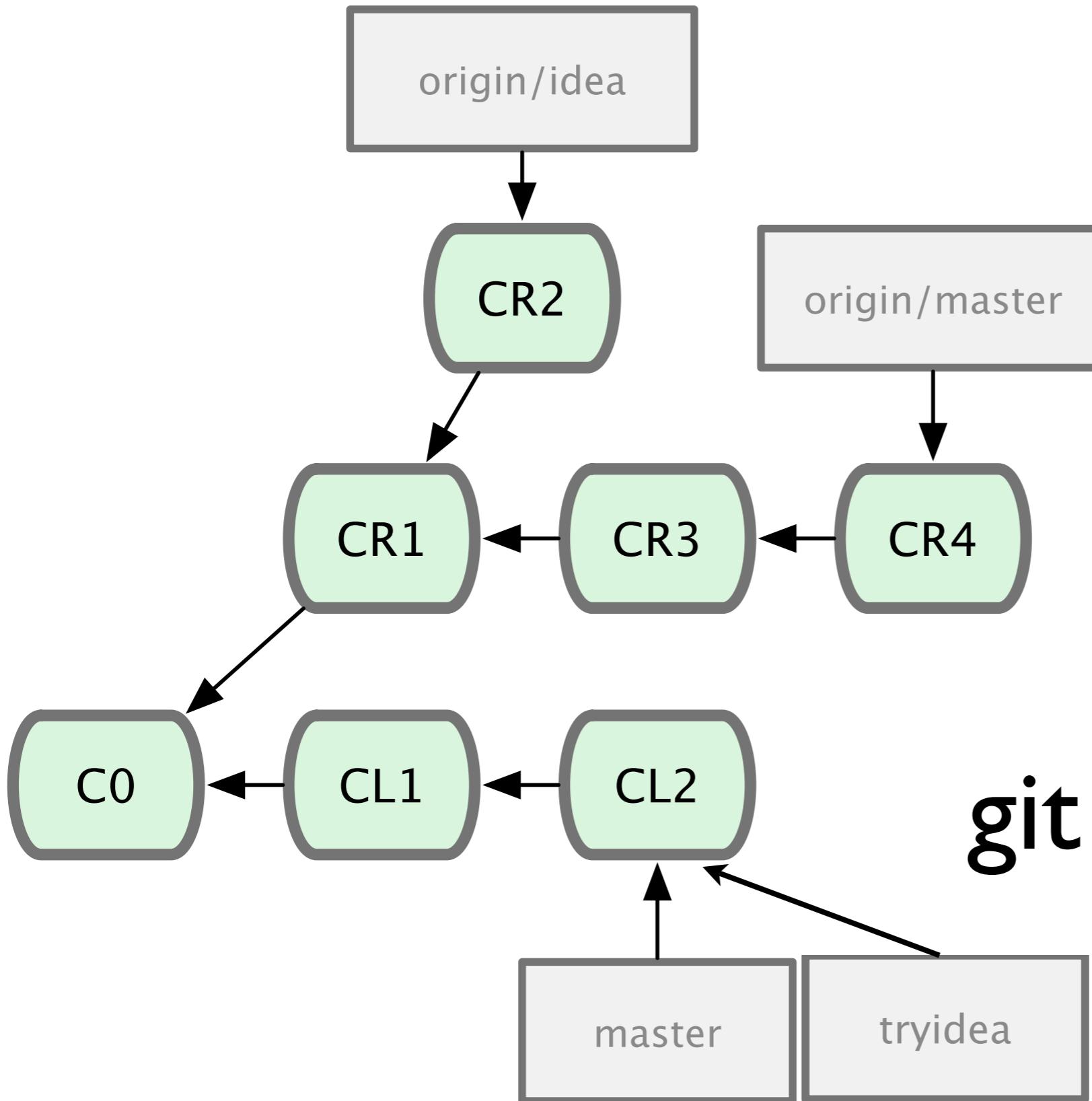




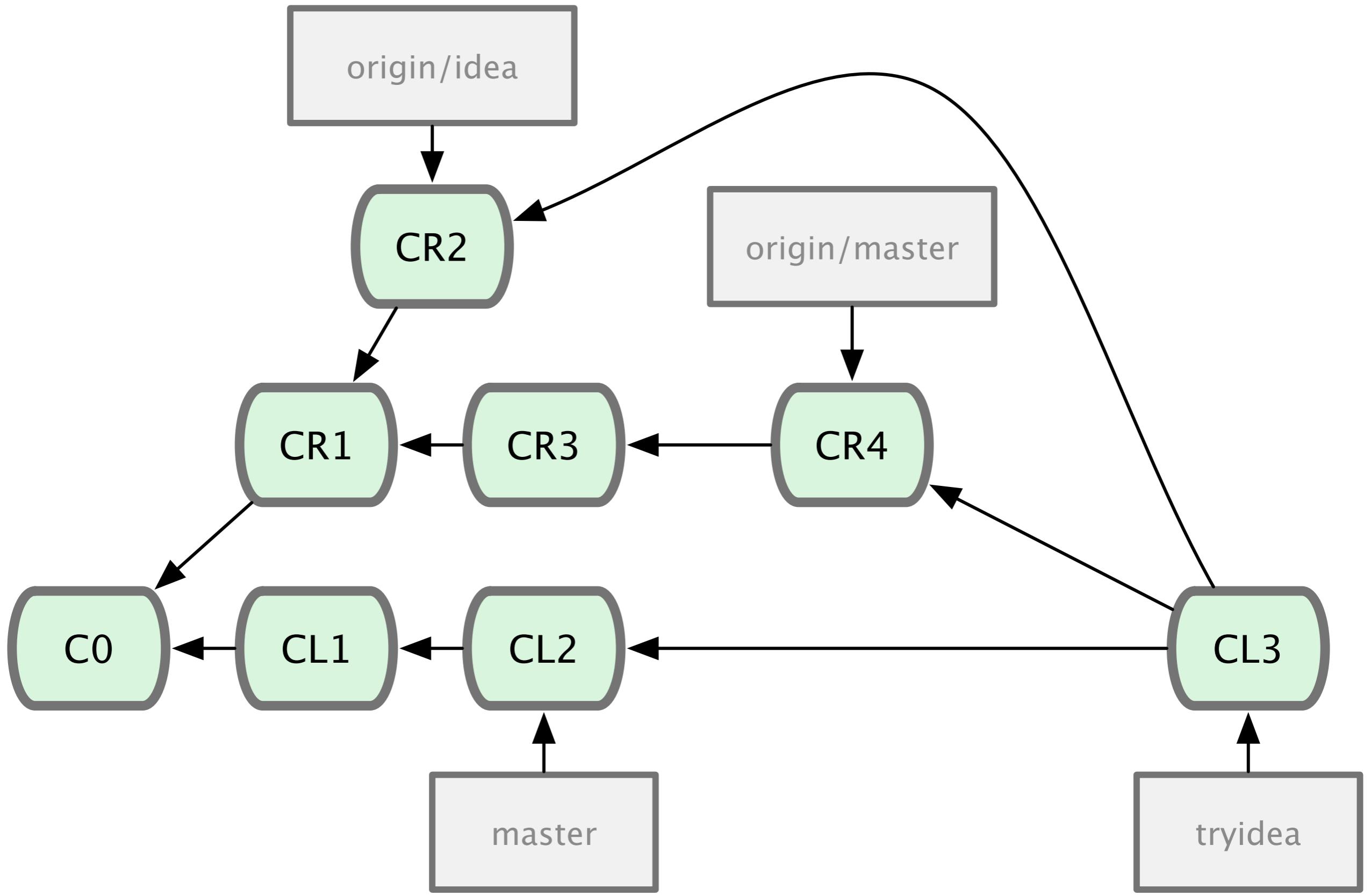
git fetch origin



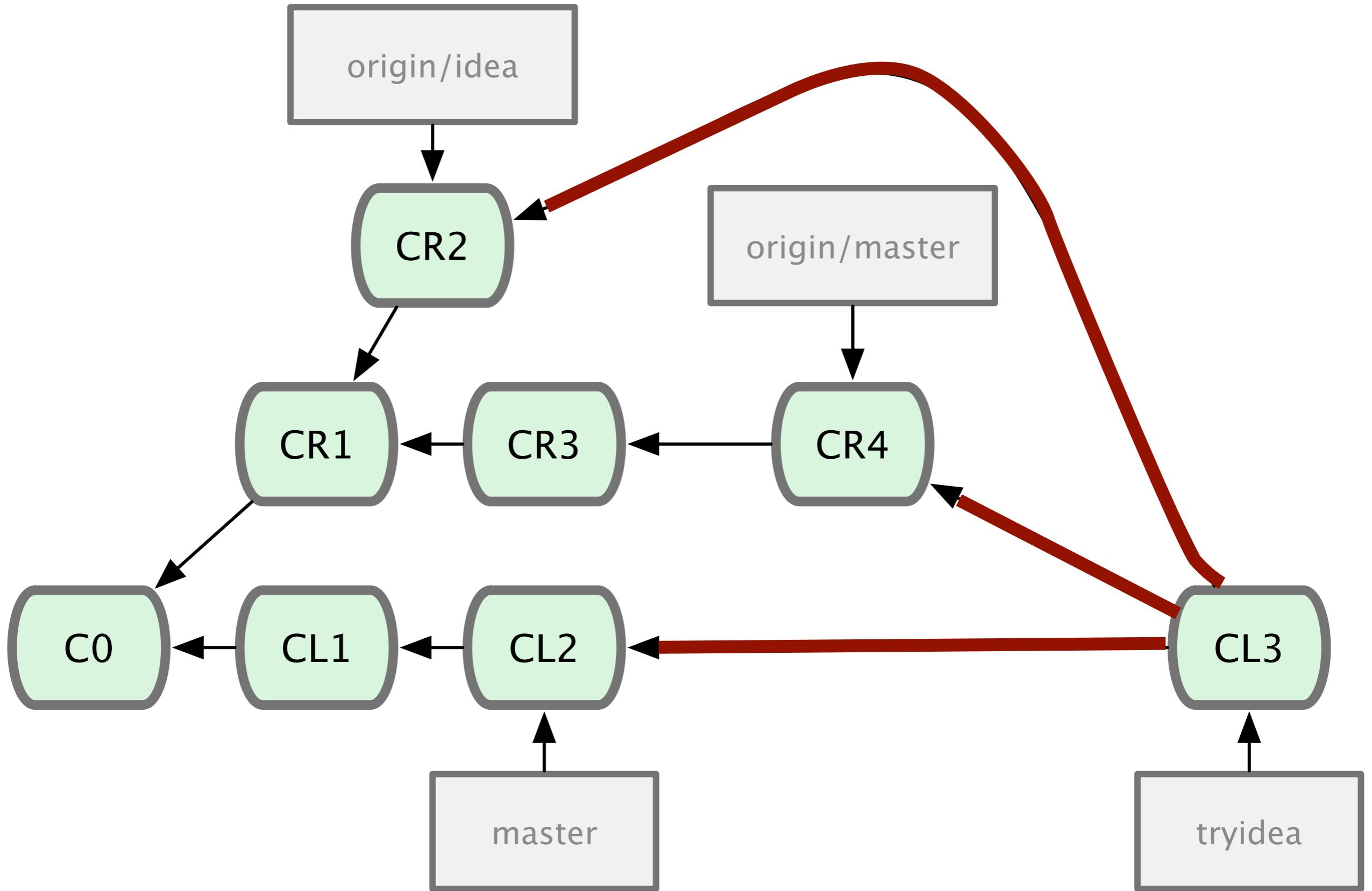
git fetch origin

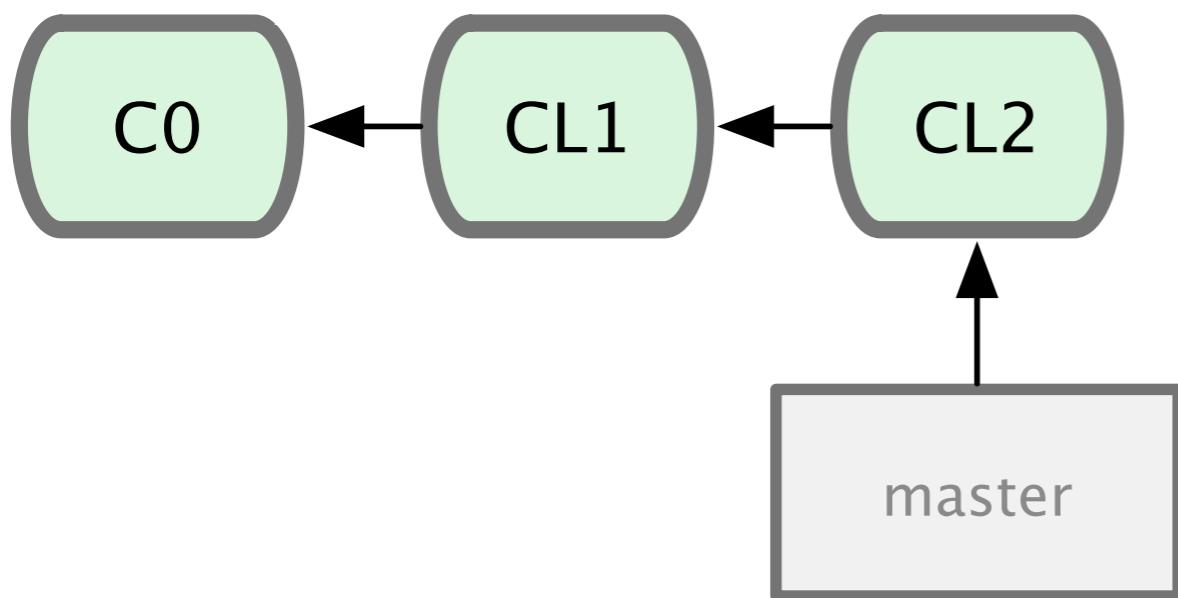


**git checkout -b
tryidea**



git merge origin/master origin/idea





remote workflow

local repo

public repo

public repo

internet

local repo

A B C

local repo

public repo

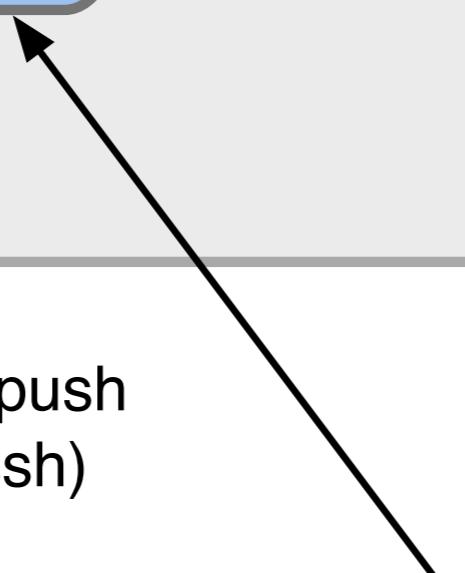
public repo

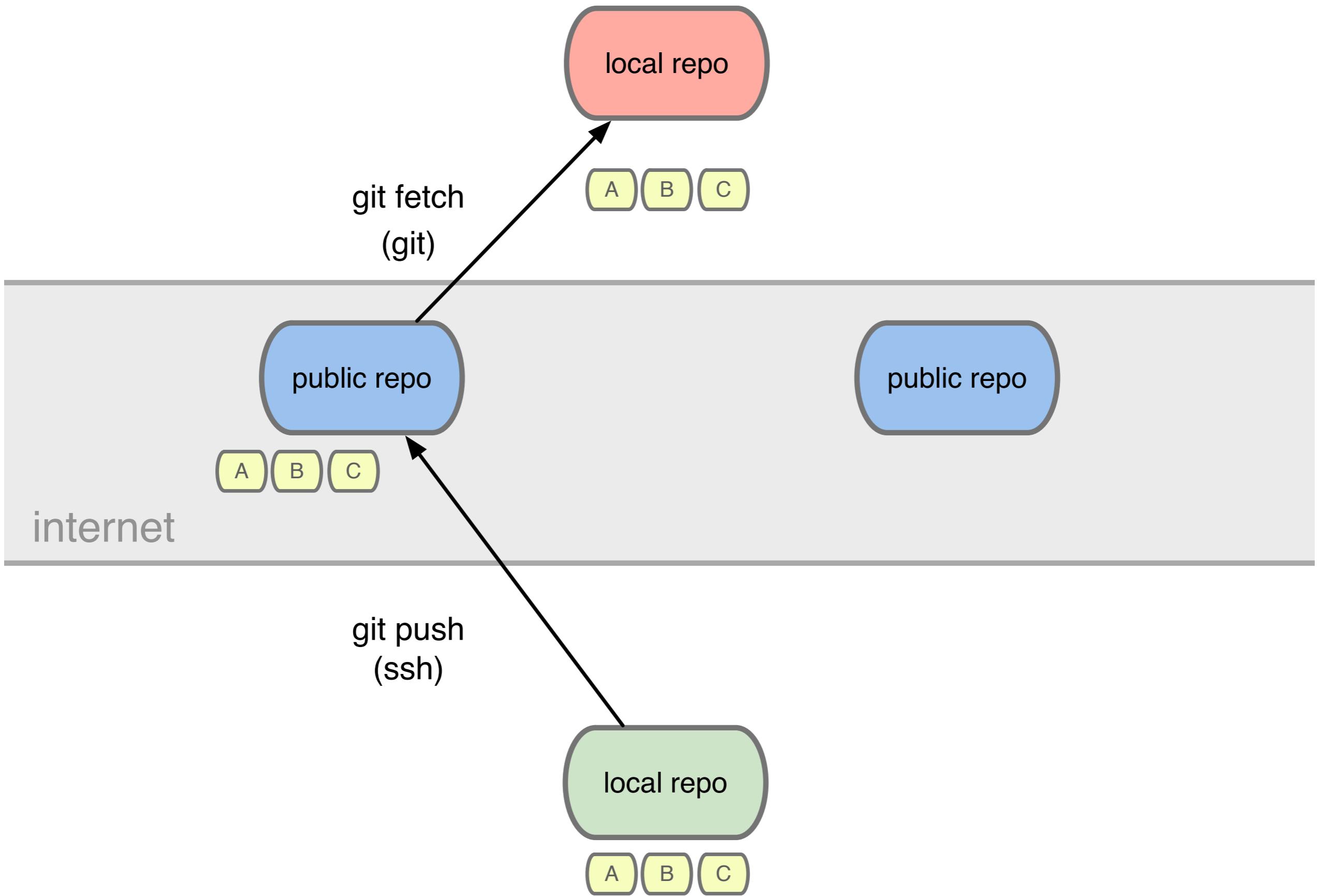
internet

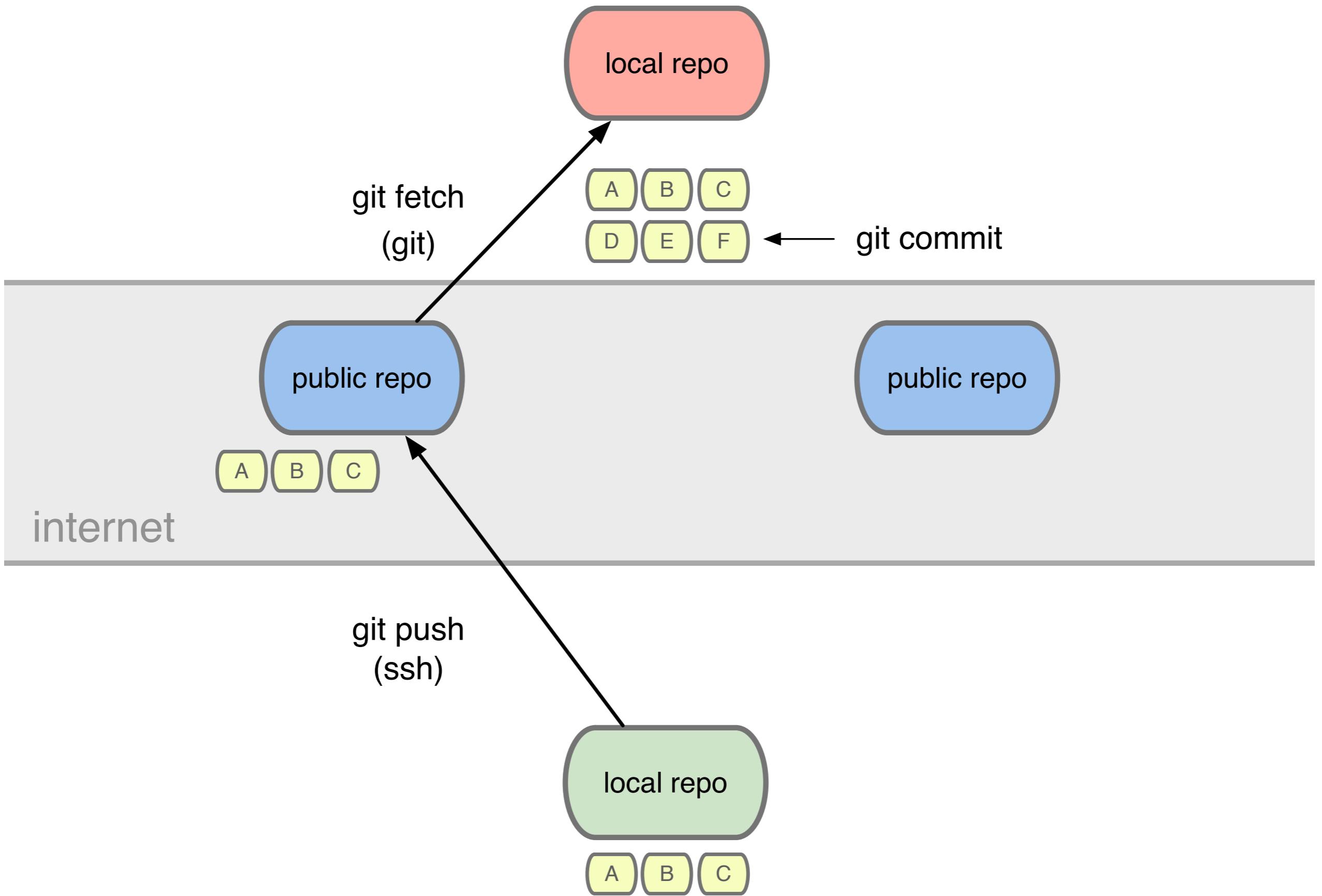
git push
(ssh)

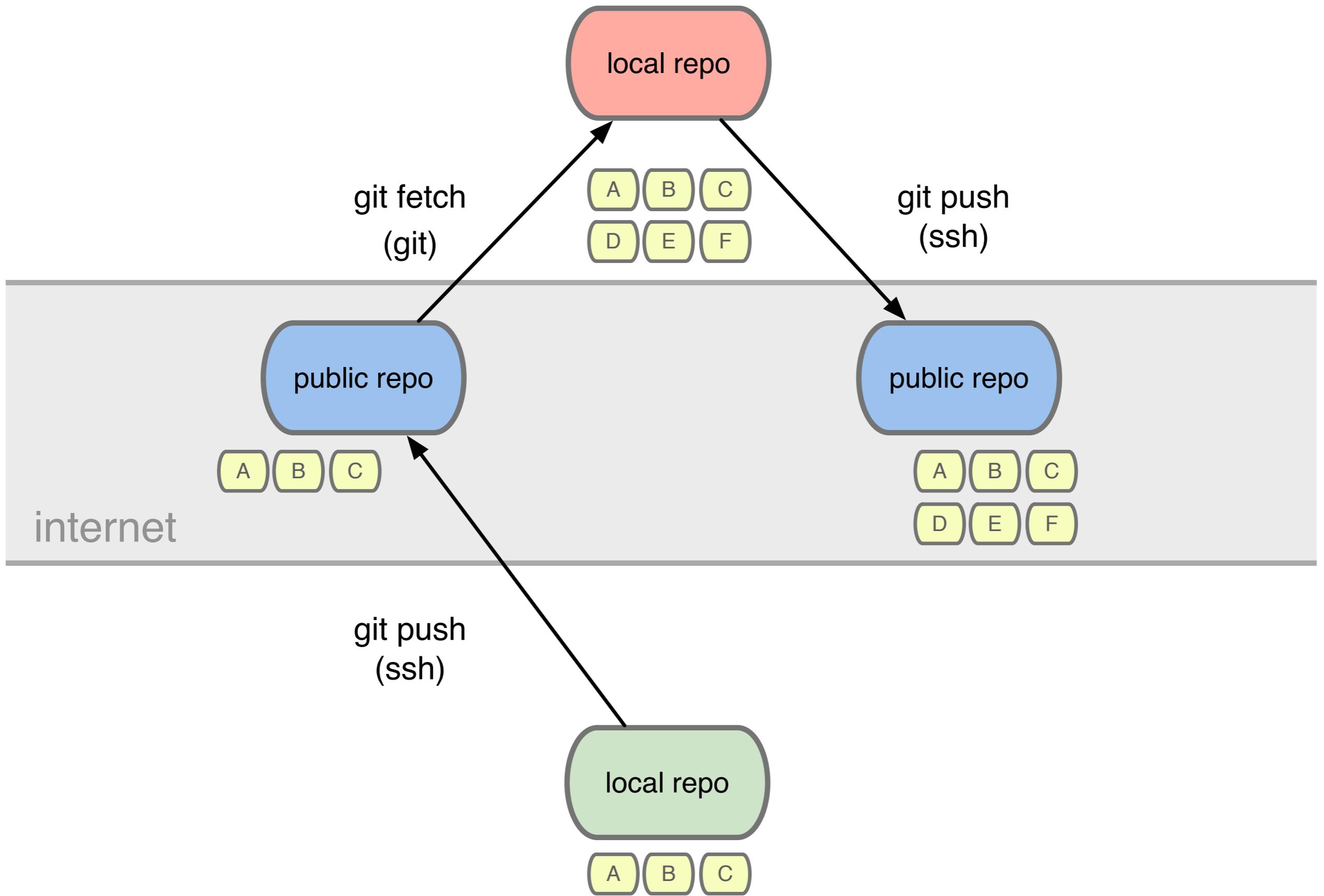
local repo

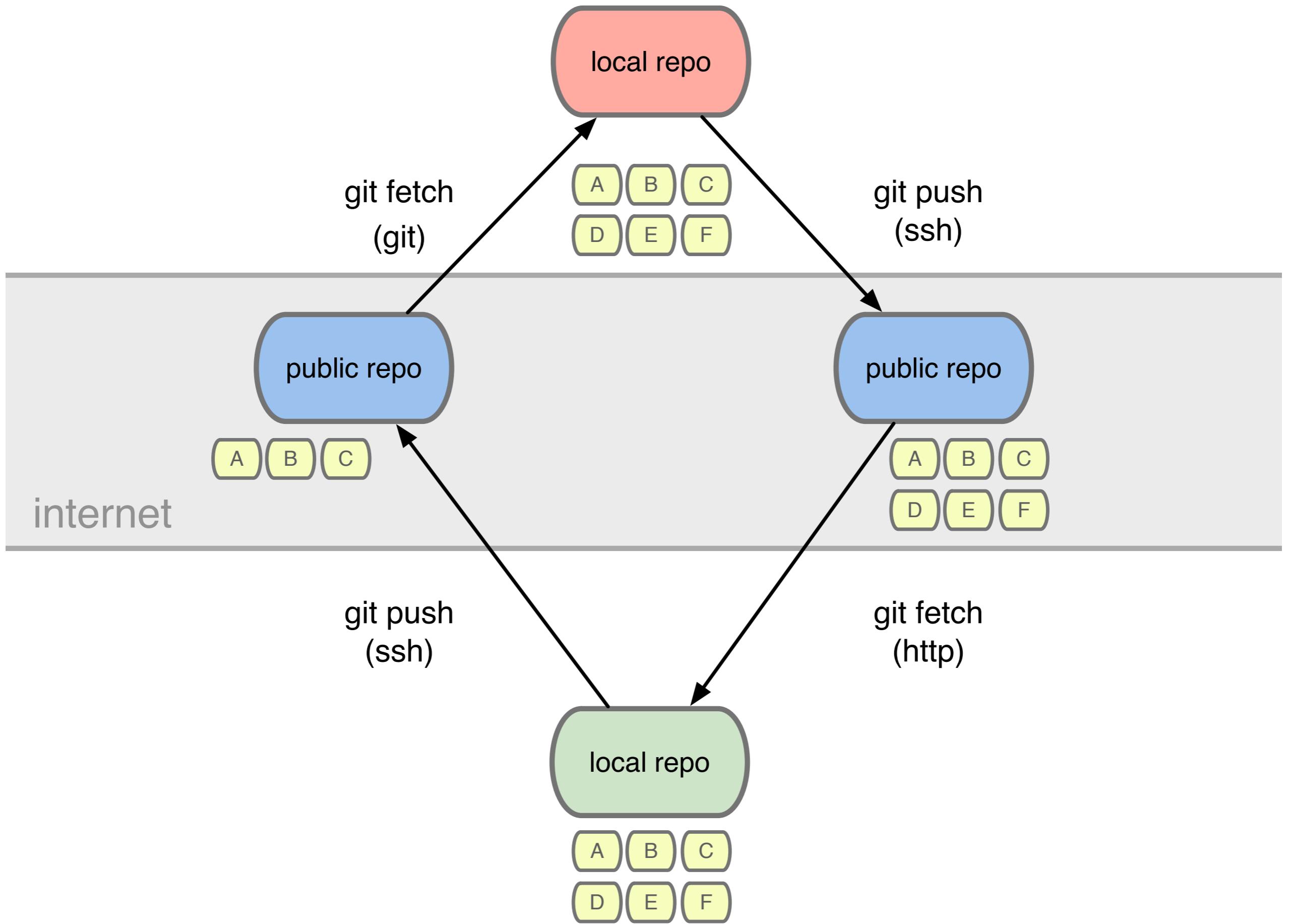
A B C

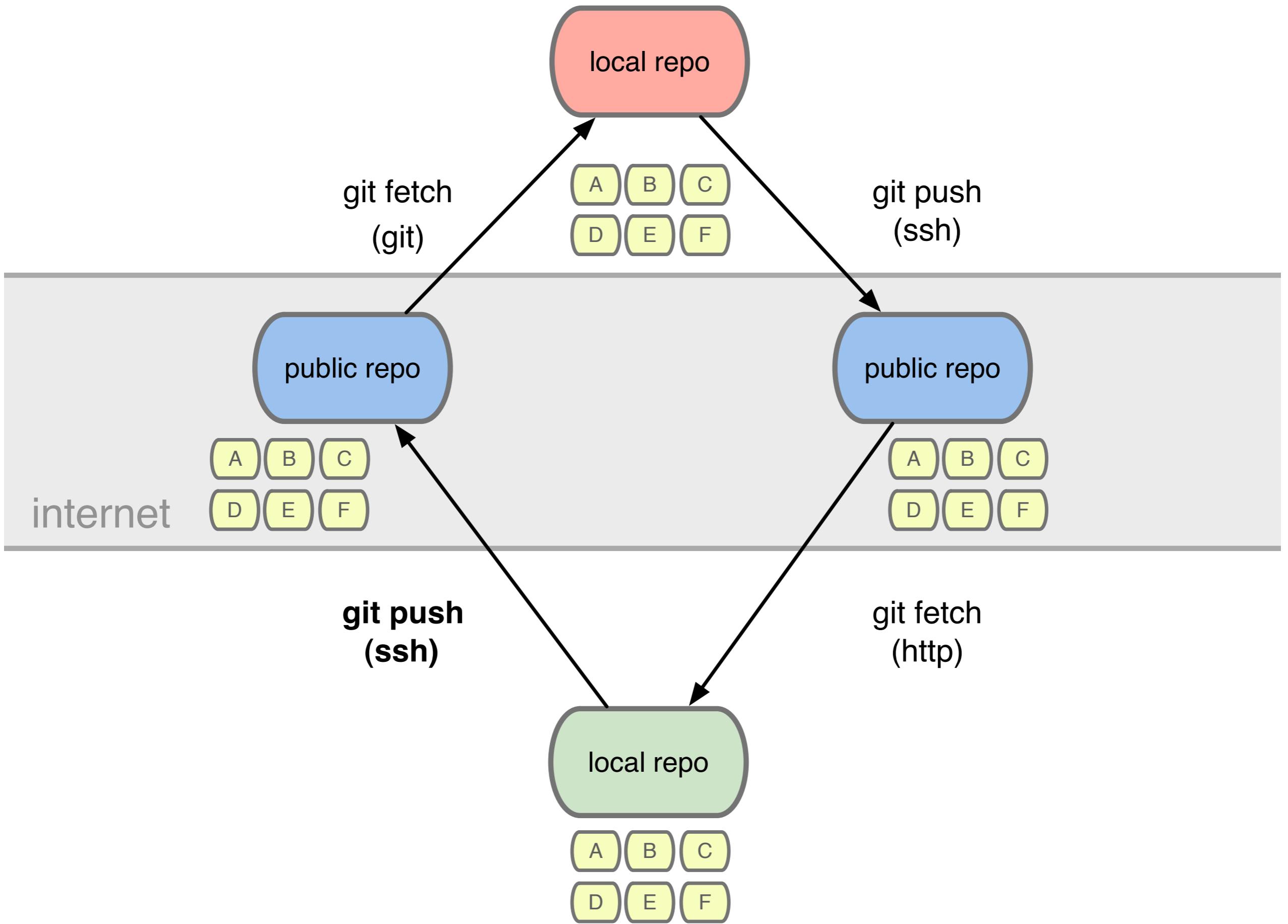








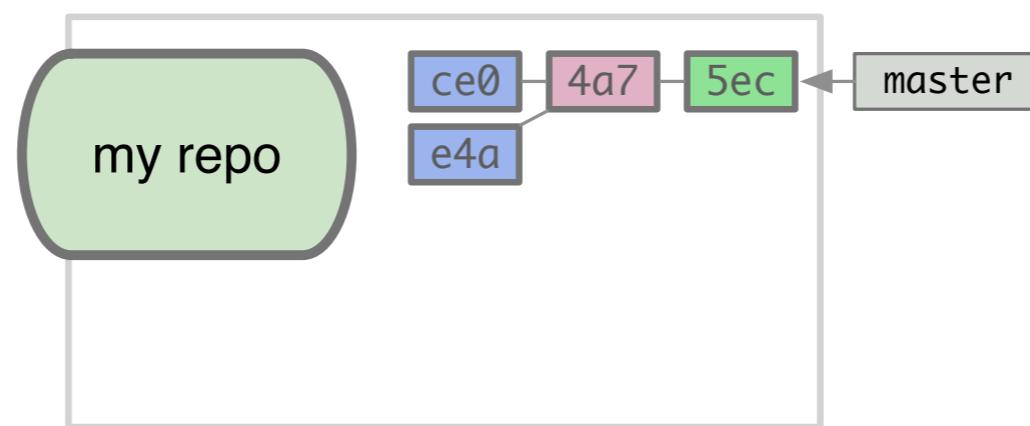




multiple remotes

developer
nick

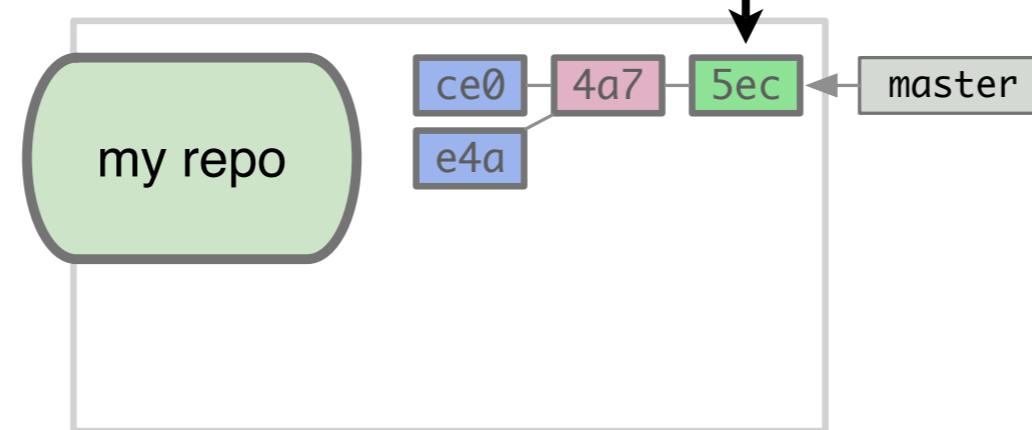
developer
jessica



developer
nick

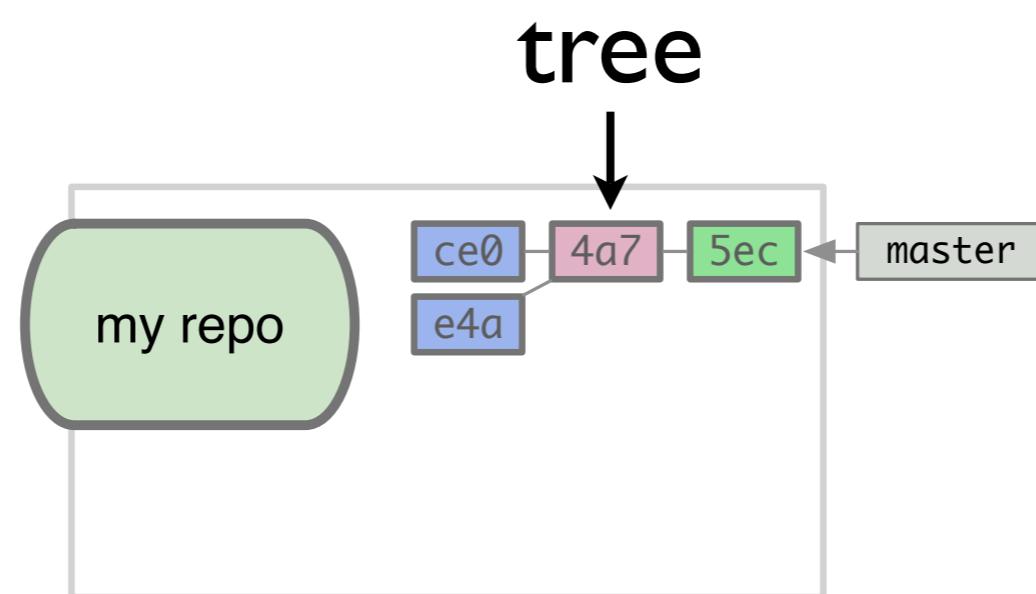
developer
jessica

commit



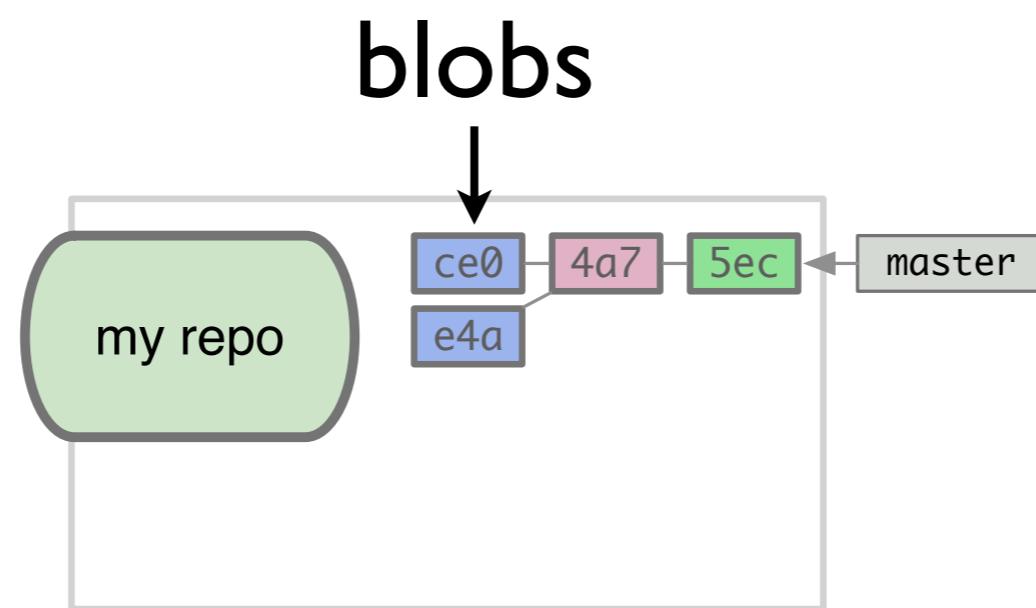
developer
nick

developer
jessica



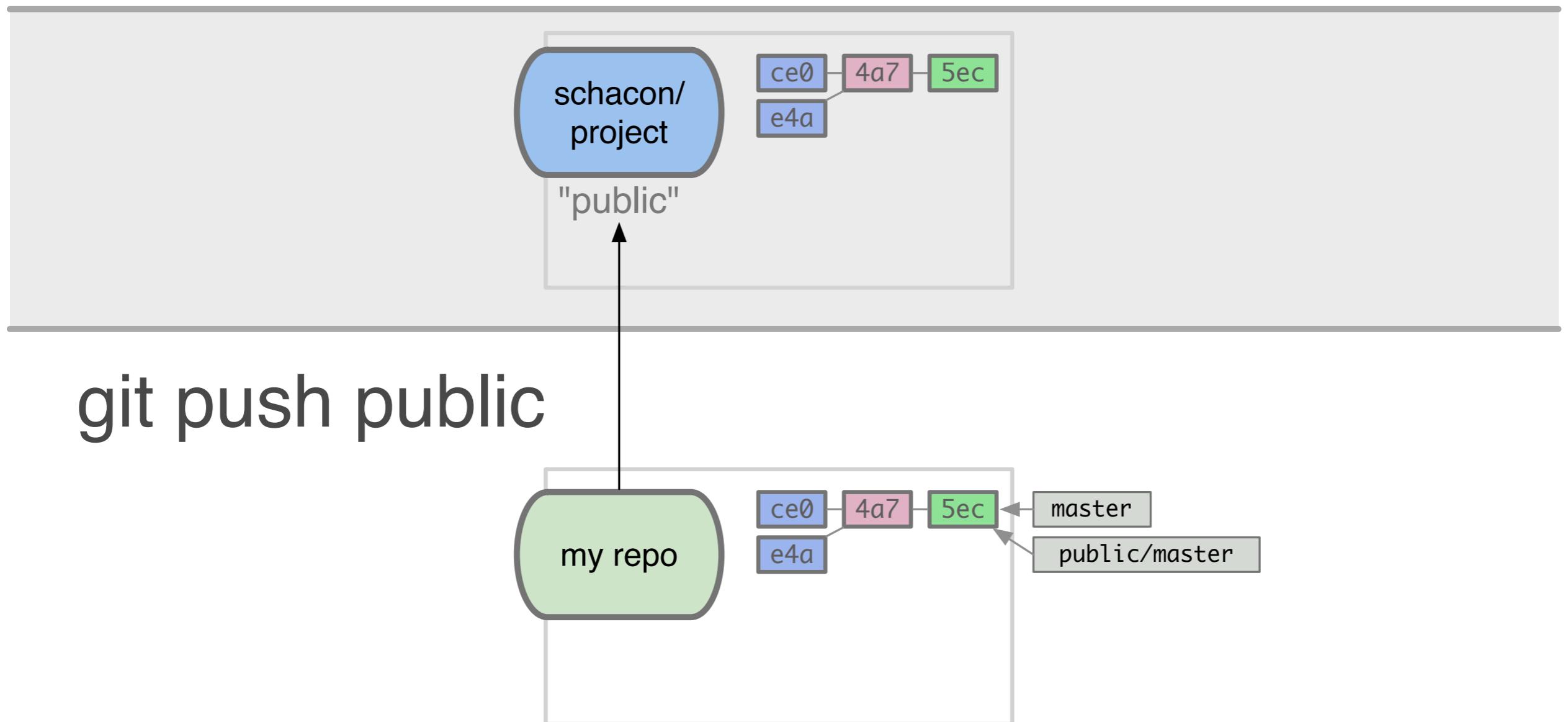
developer
nick

developer
jessica

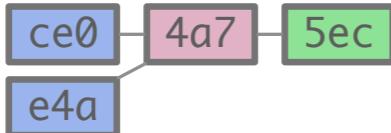


developer
nick

developer
jessica



developer
nick

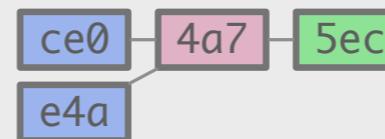


developer
jessica

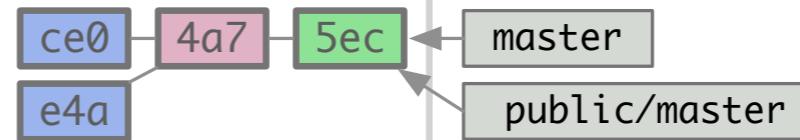
git clone (url)

schacon/
project

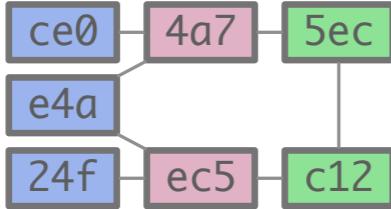
"public"



my repo



developer
nick

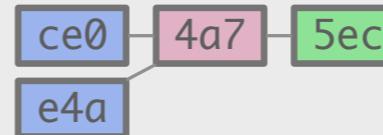


developer
jessica

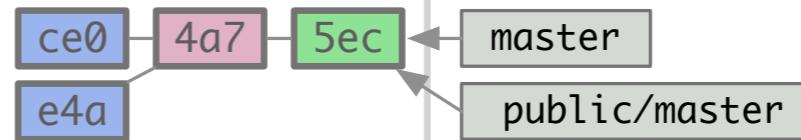
git commit

schacon/
project

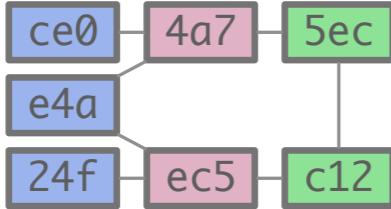
"public"



my repo

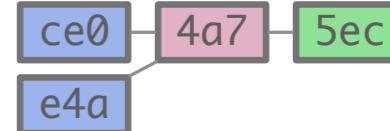


developer
nick



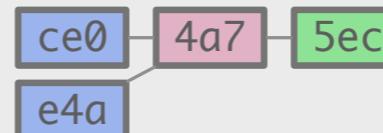
git clone (url)

developer
jessica

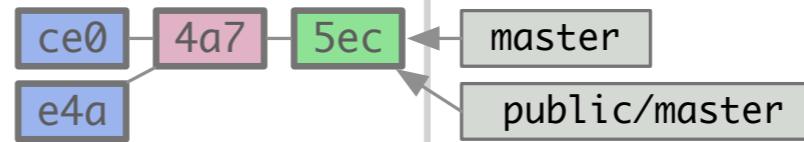


schacon/
project

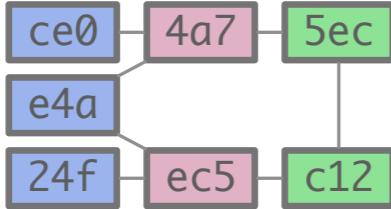
"public"



my repo

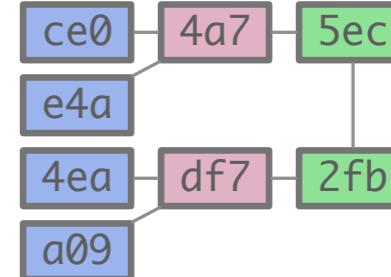


developer
nick



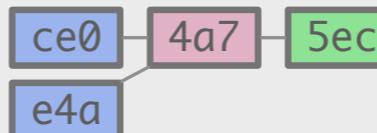
git commit

developer
jessica

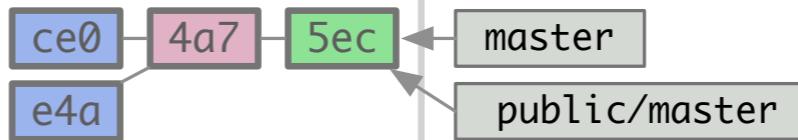


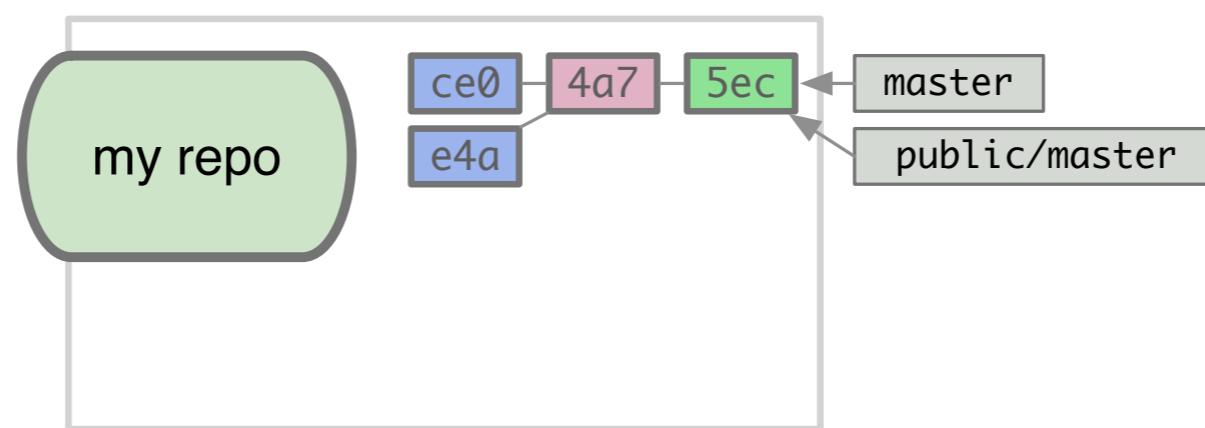
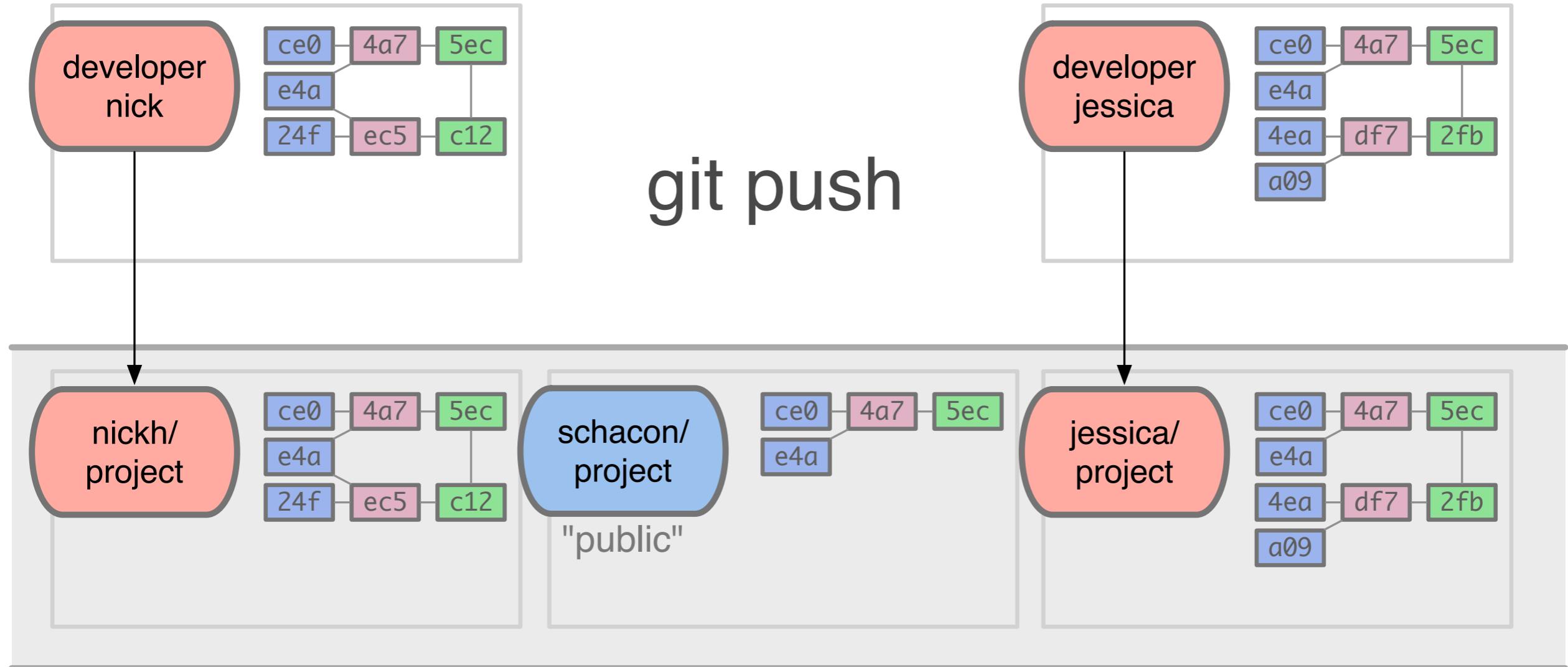
schacon/
project

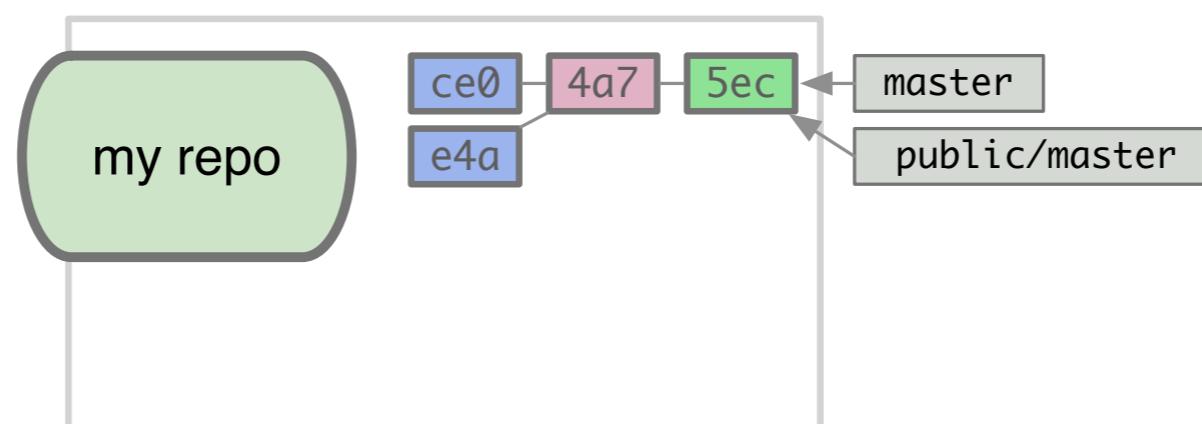
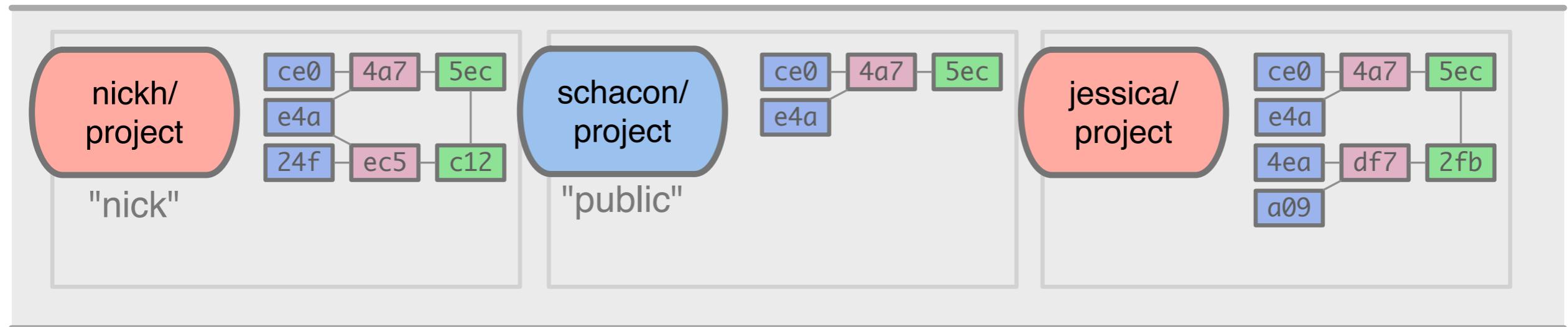
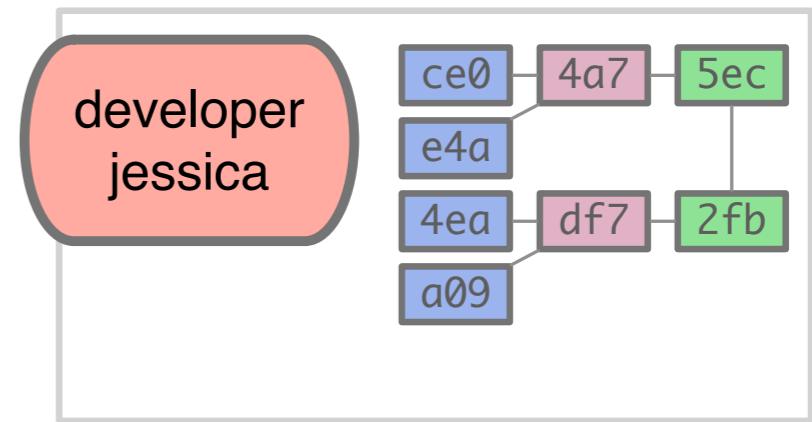
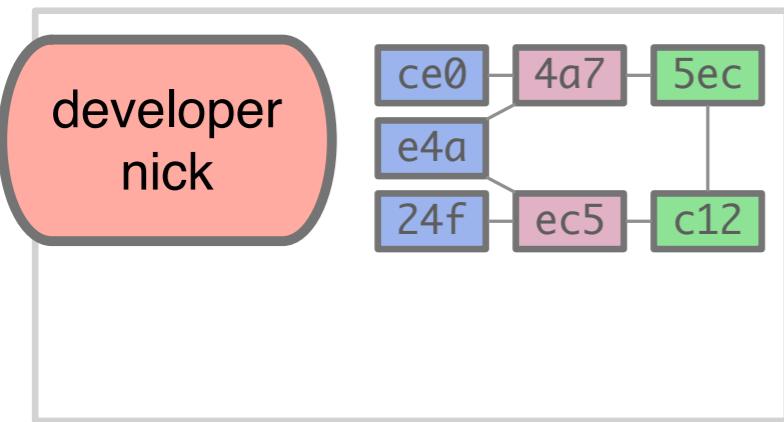
"public"



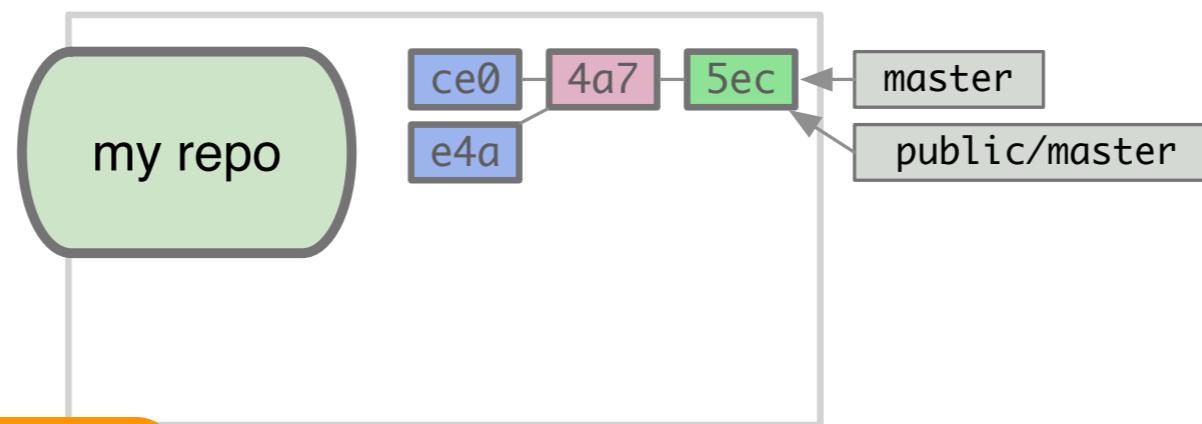
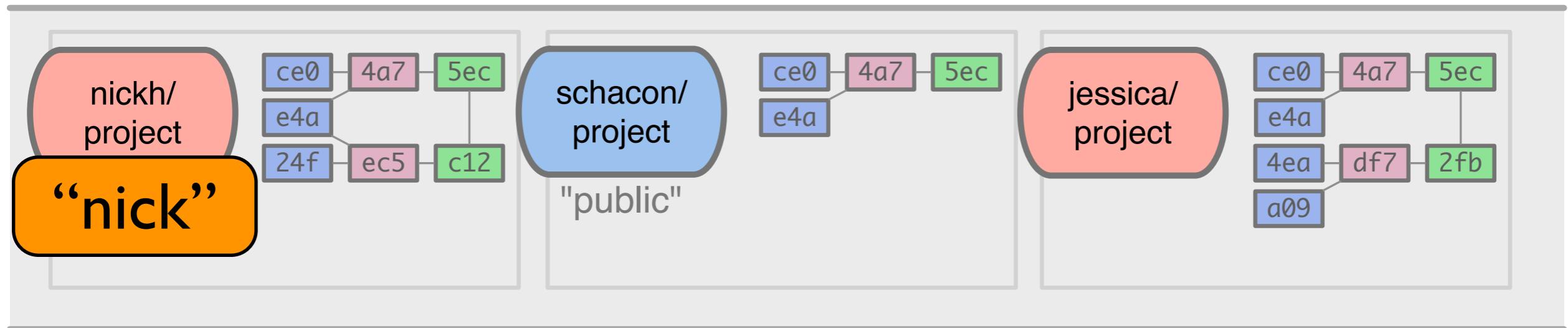
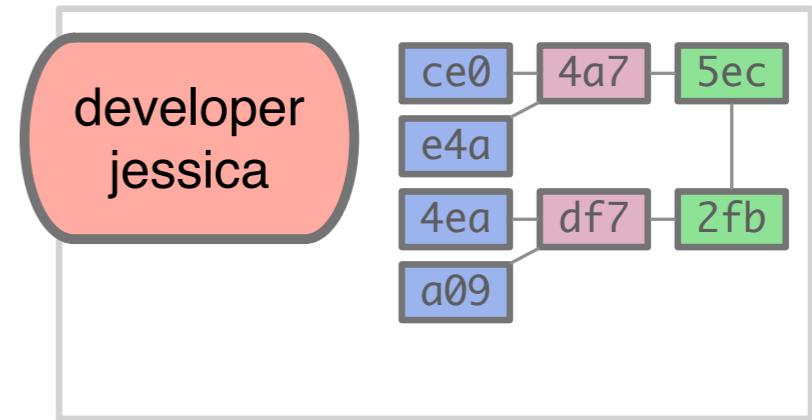
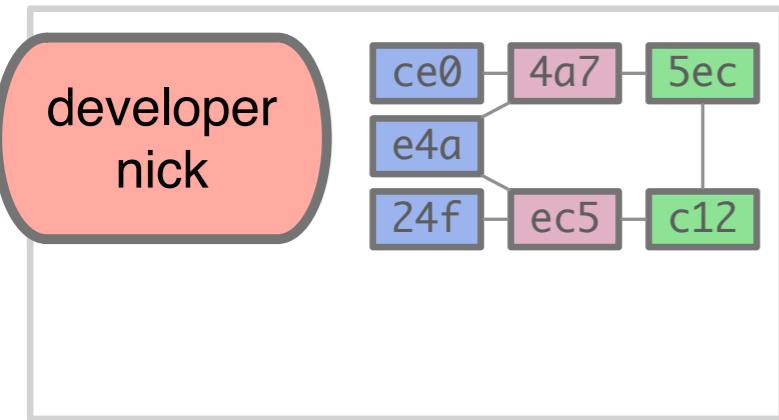
my repo



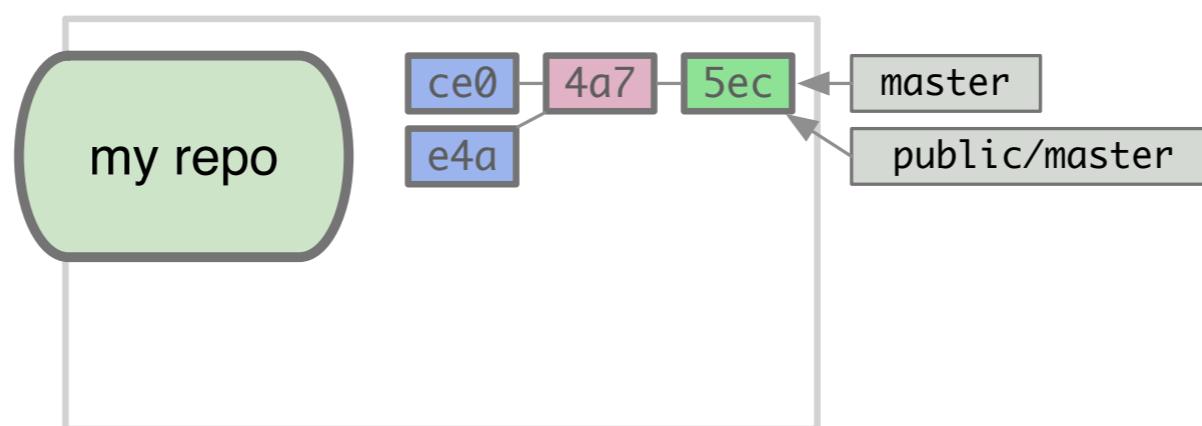
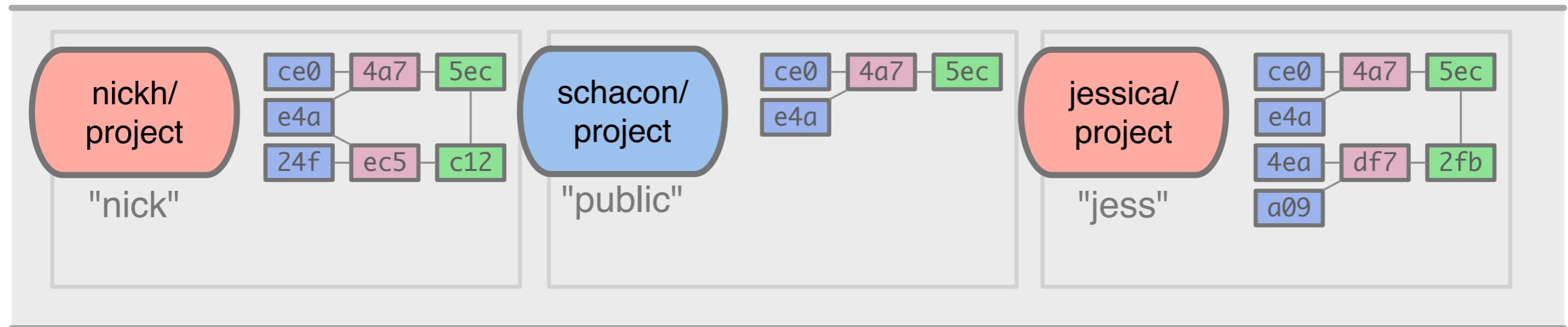
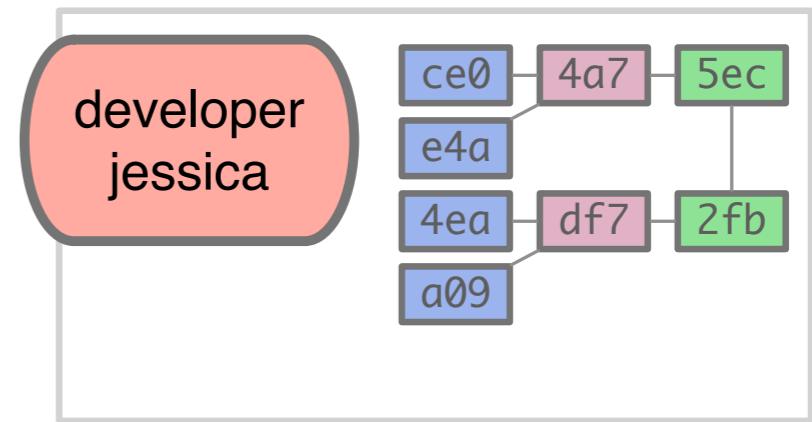
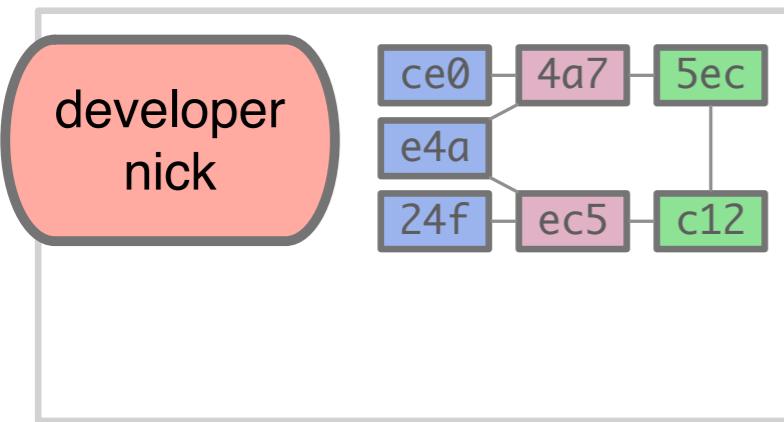




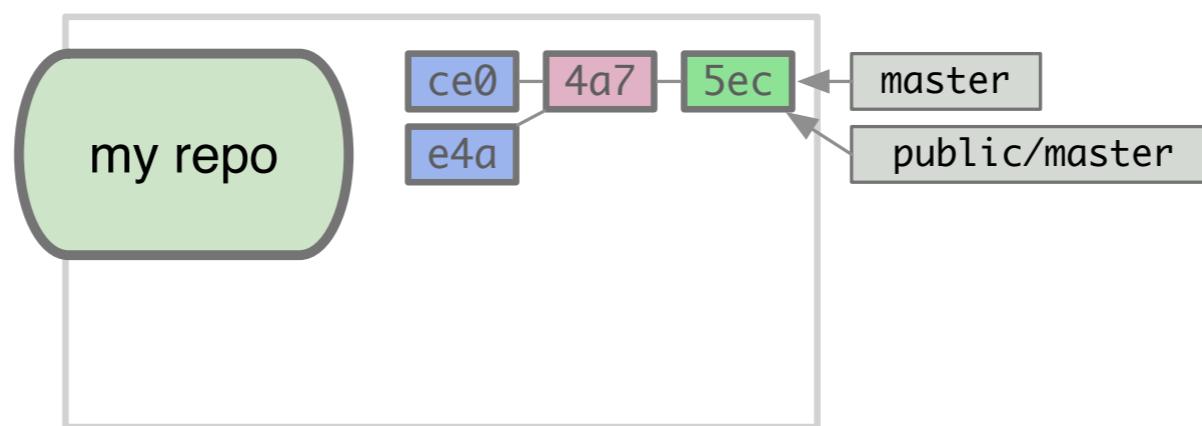
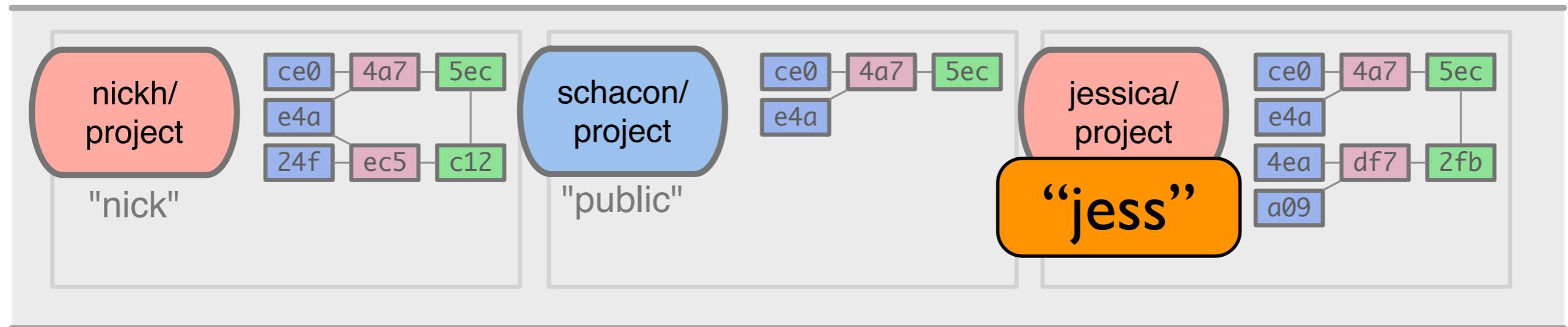
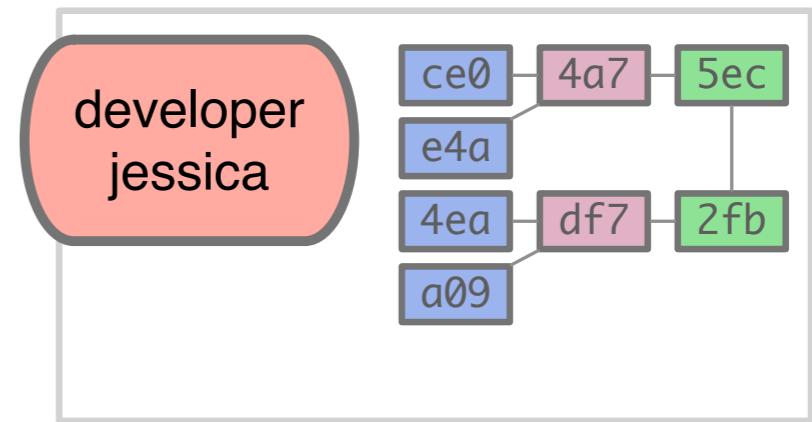
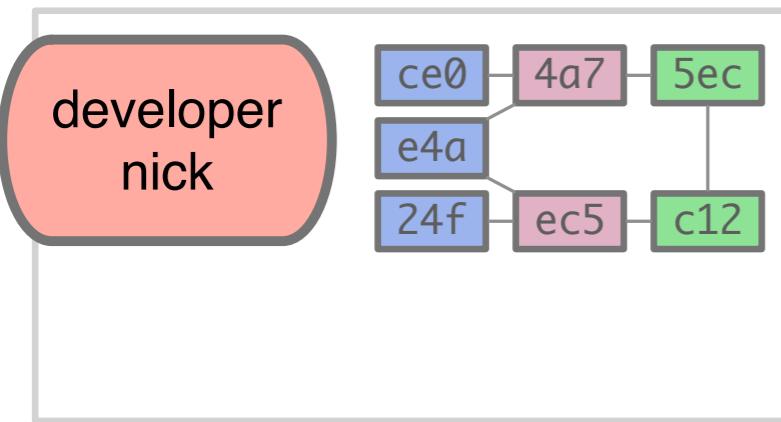
`git remote add nick git://github.com/nickh/project.git`



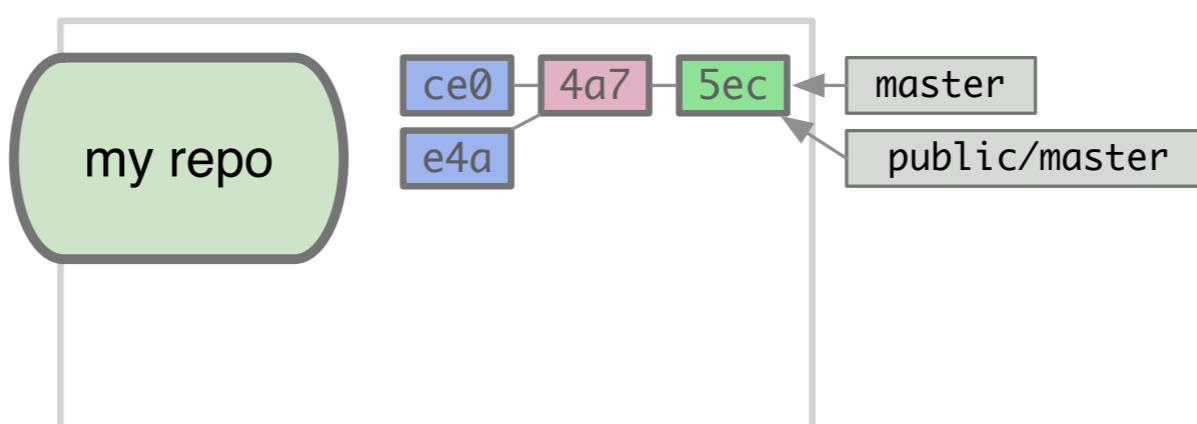
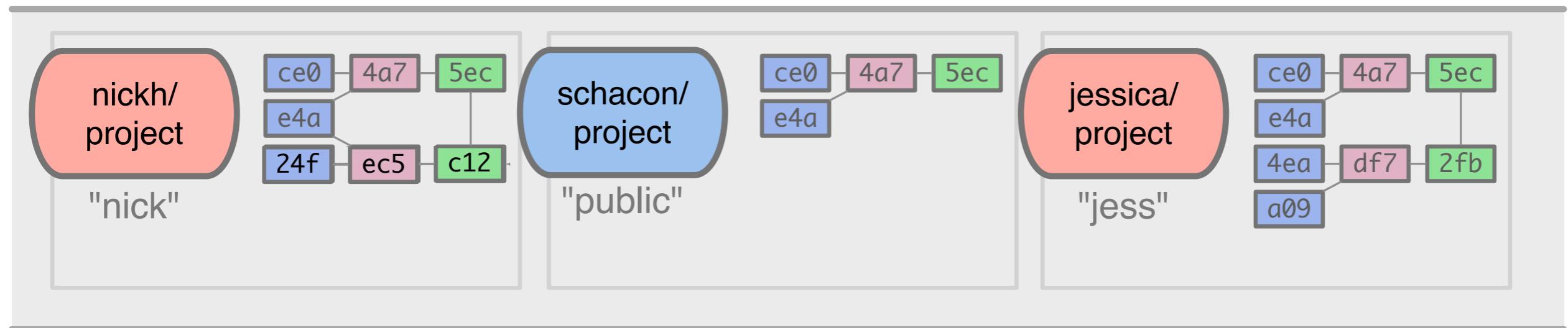
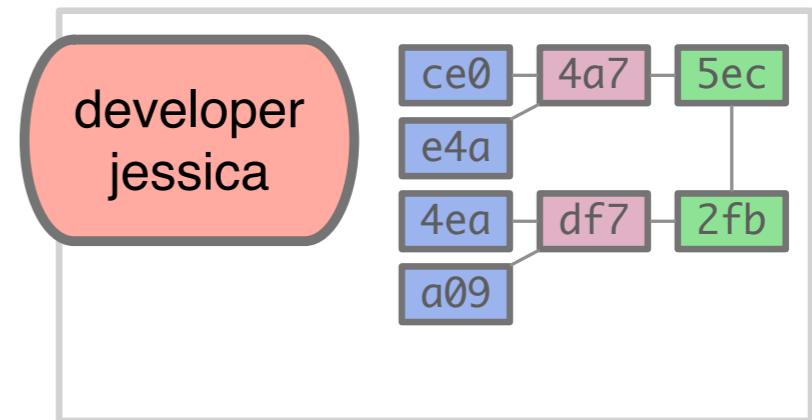
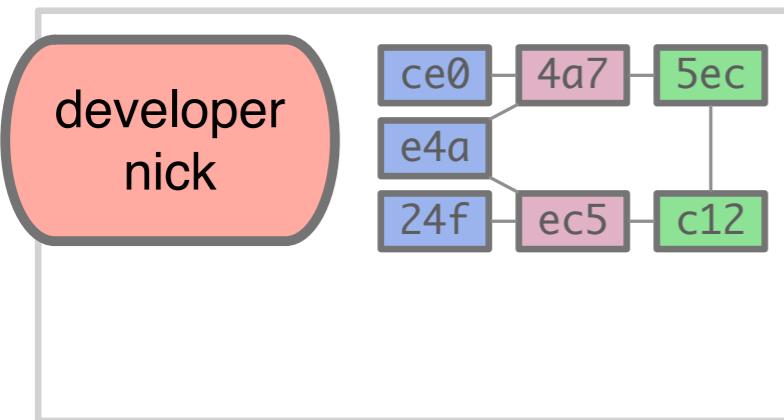
`git remote add nick git://github.com/nickh/project.git`



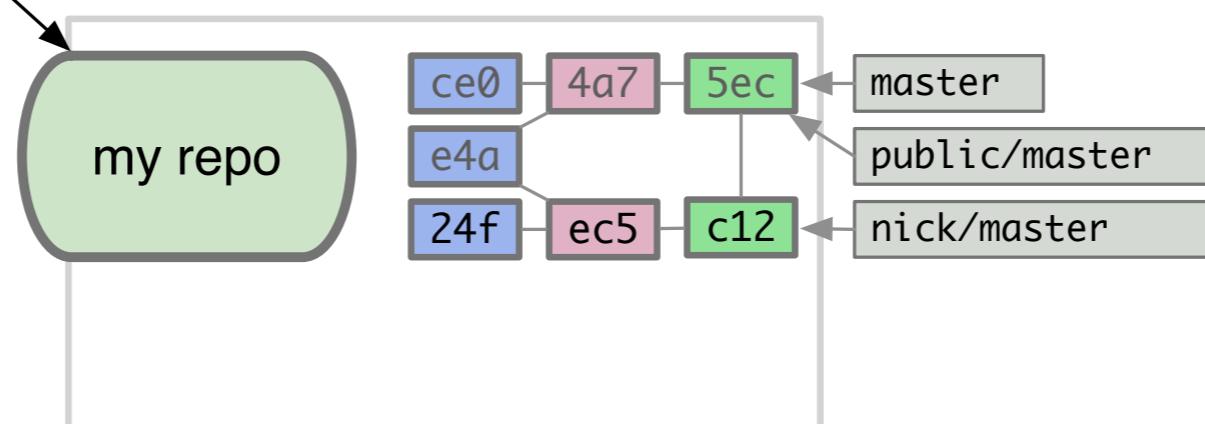
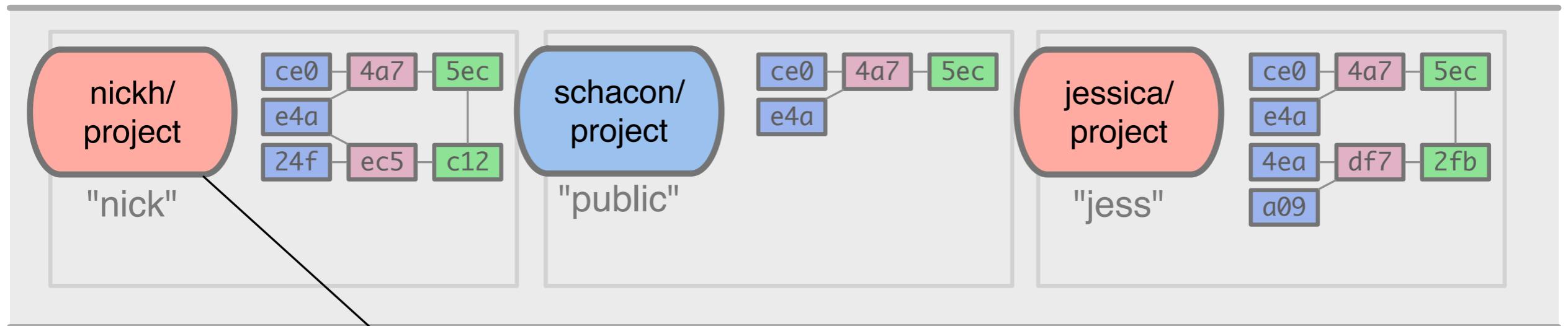
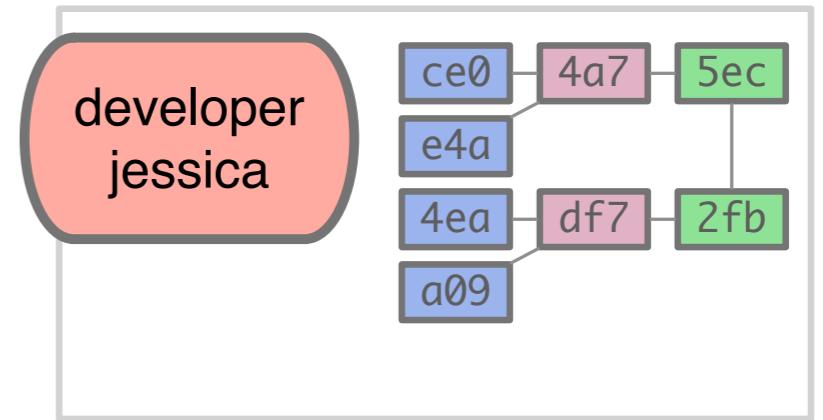
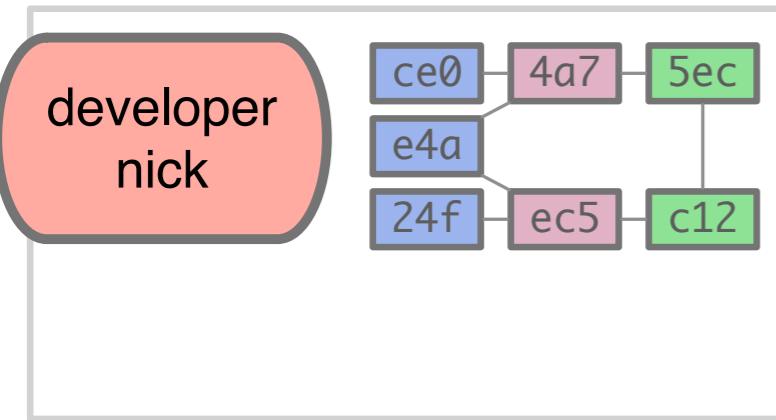
`git remote add jess git://github.com/jessica/project.git`

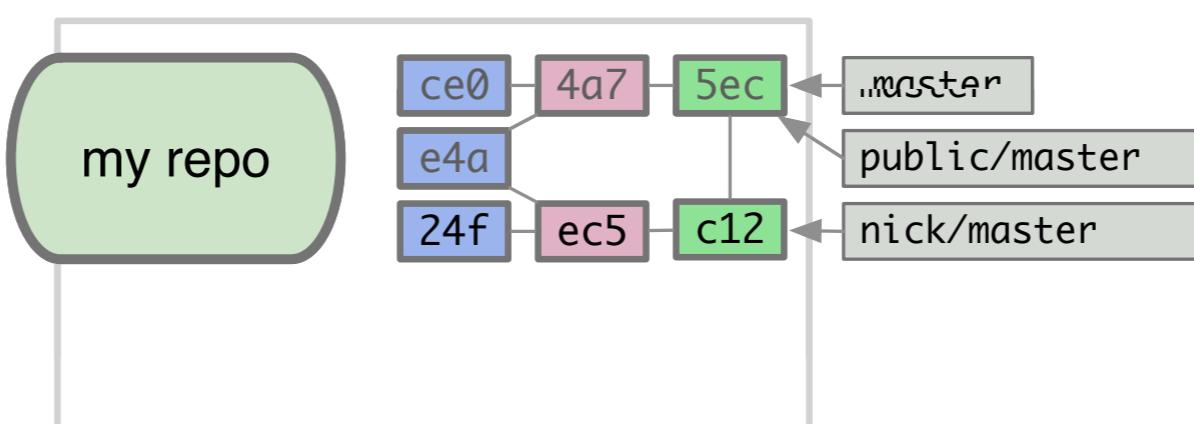
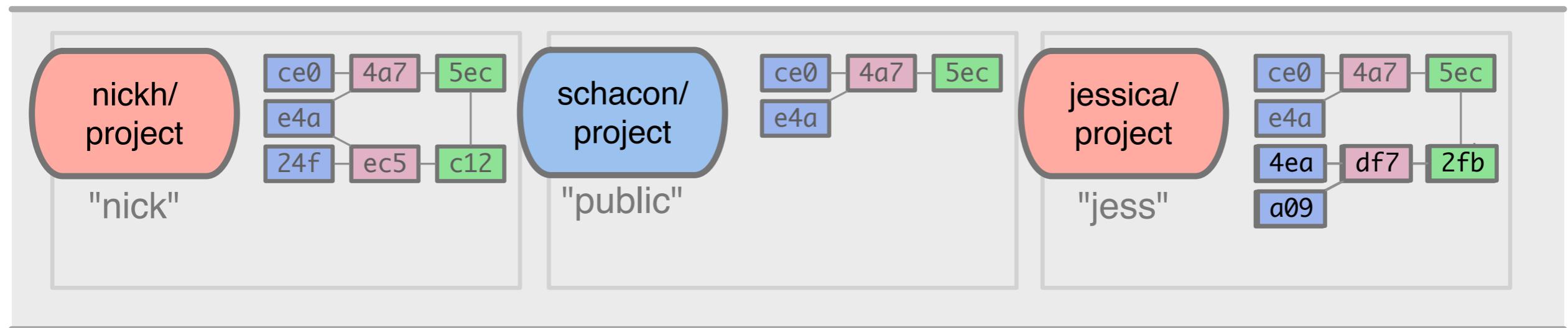
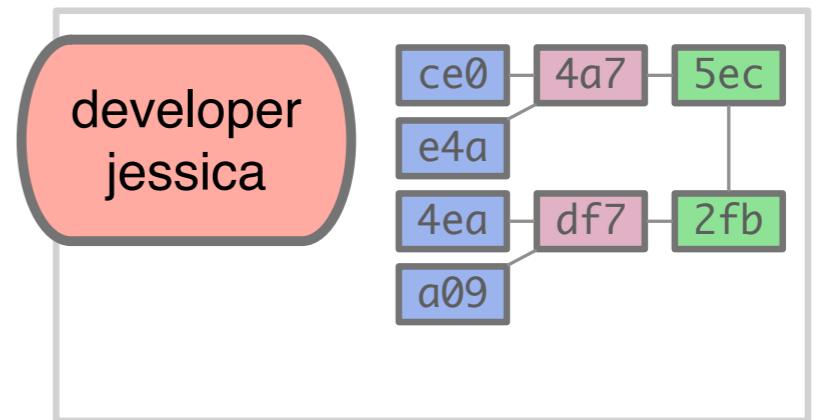
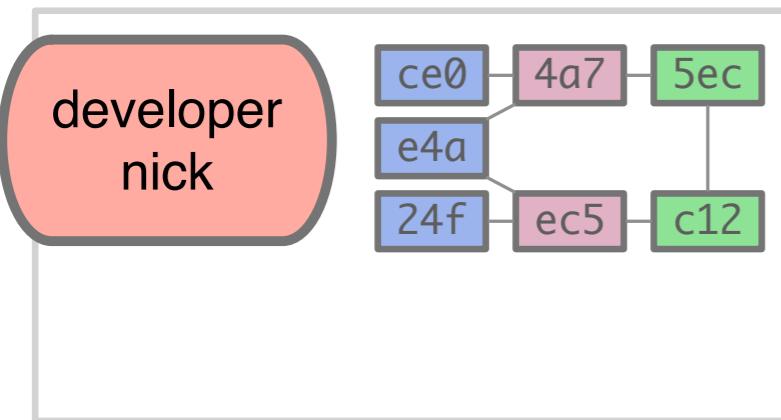


`git remote add jess git://github.com/jessica/project.git`

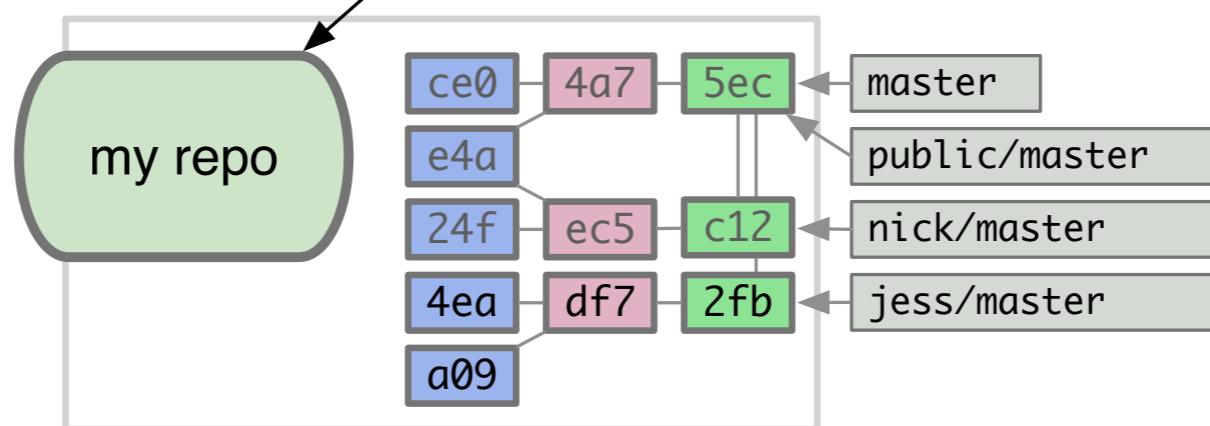
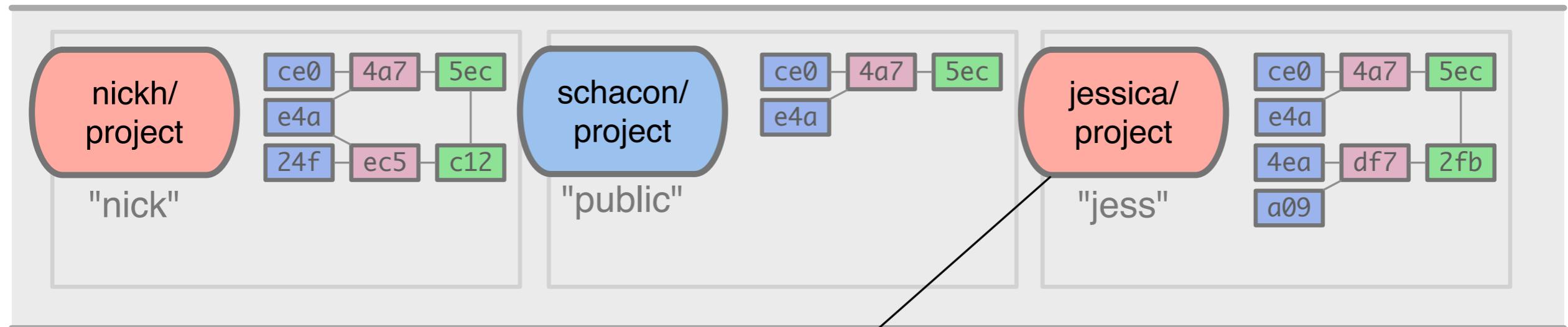
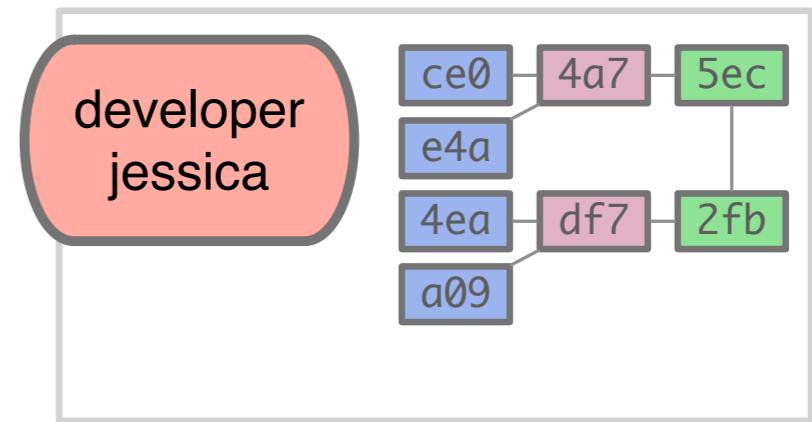
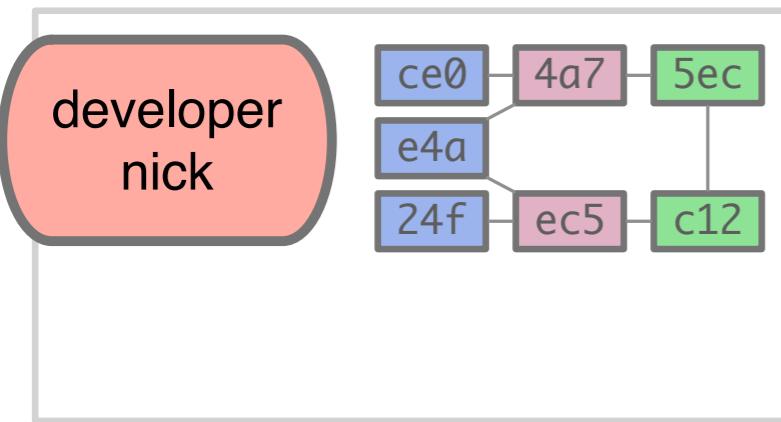


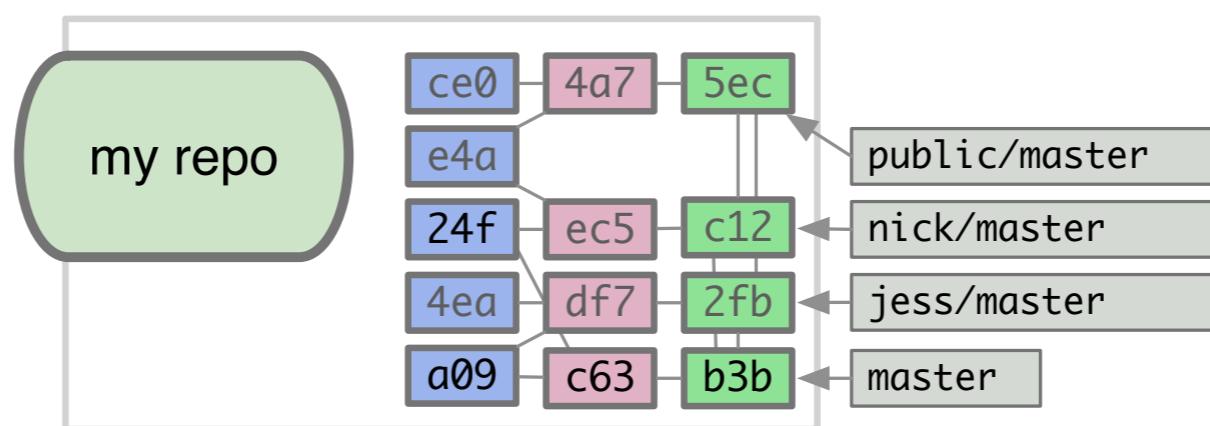
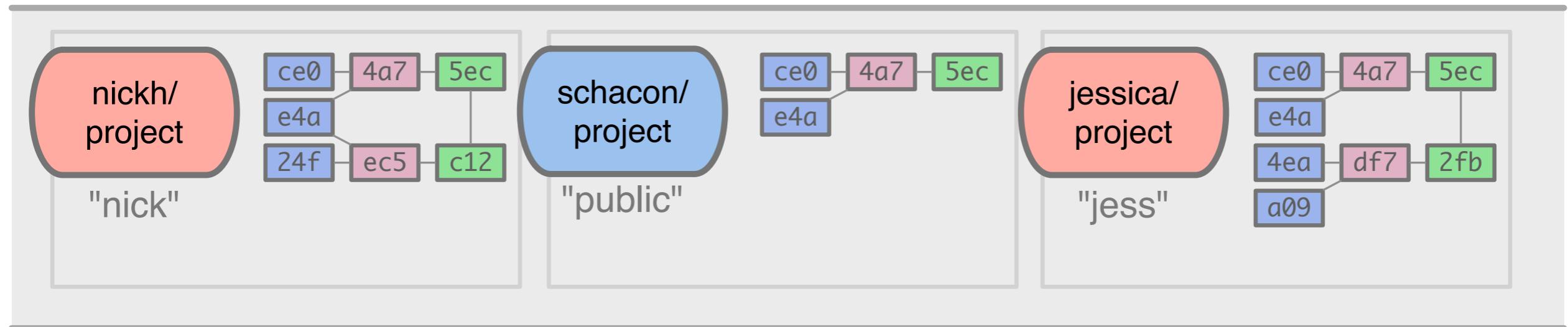
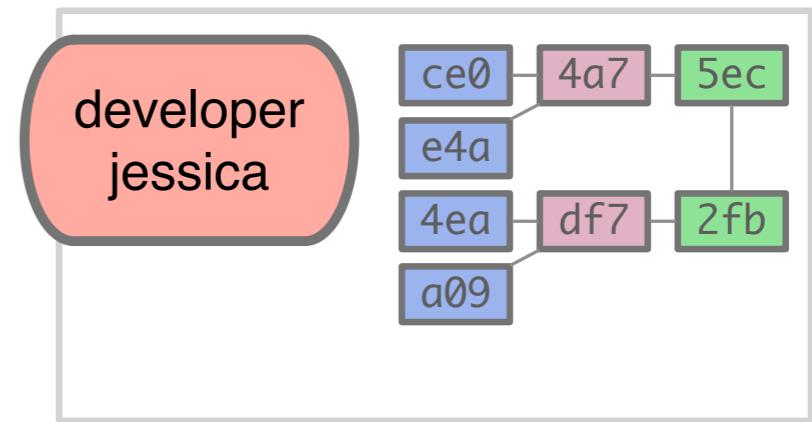
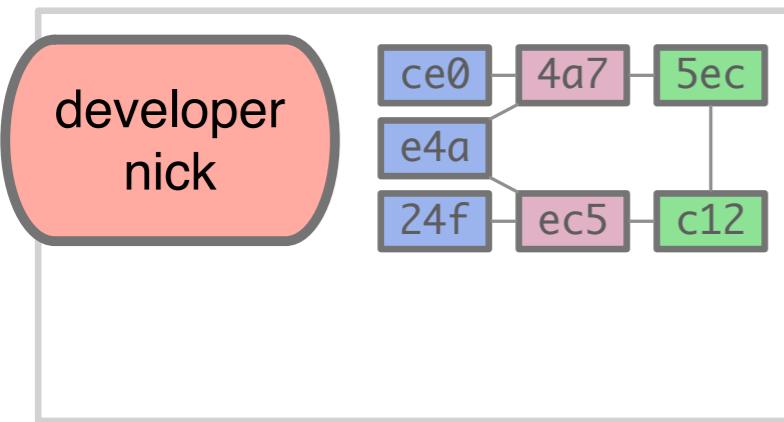
git fetch nick



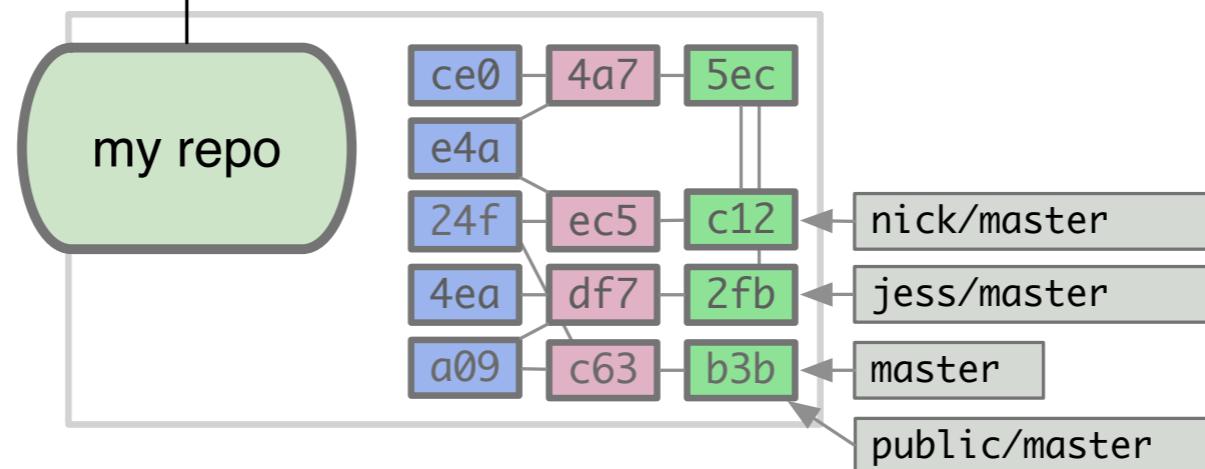
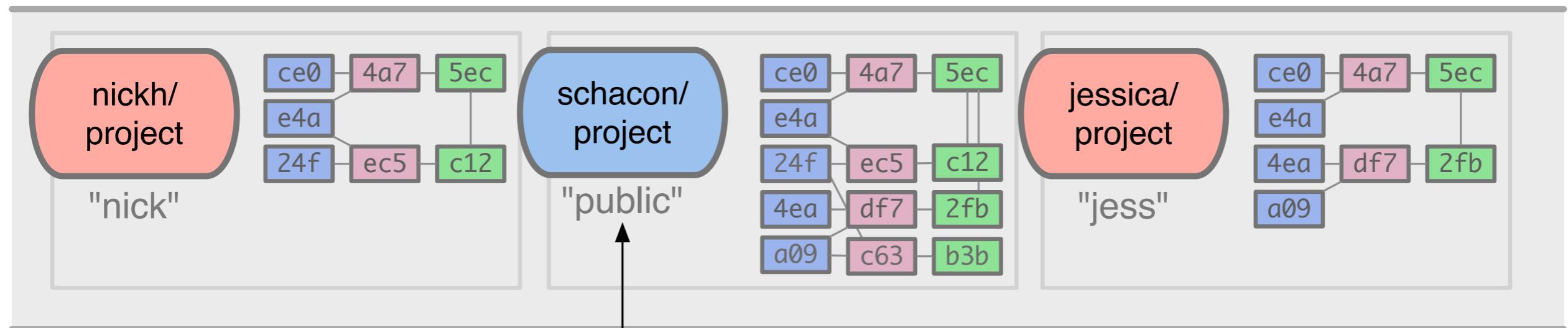
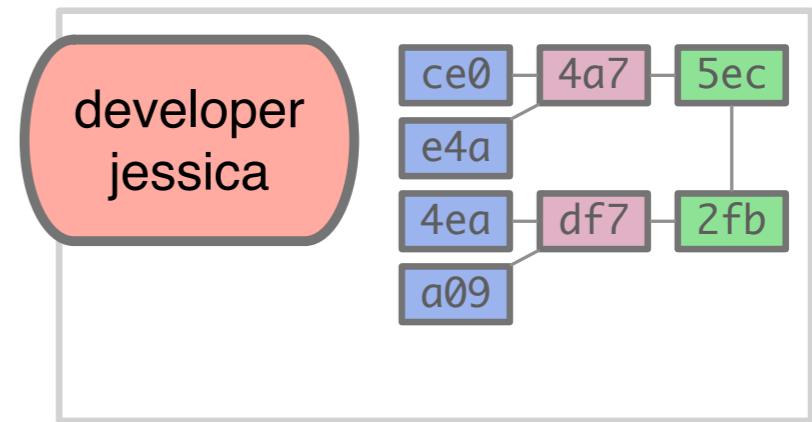
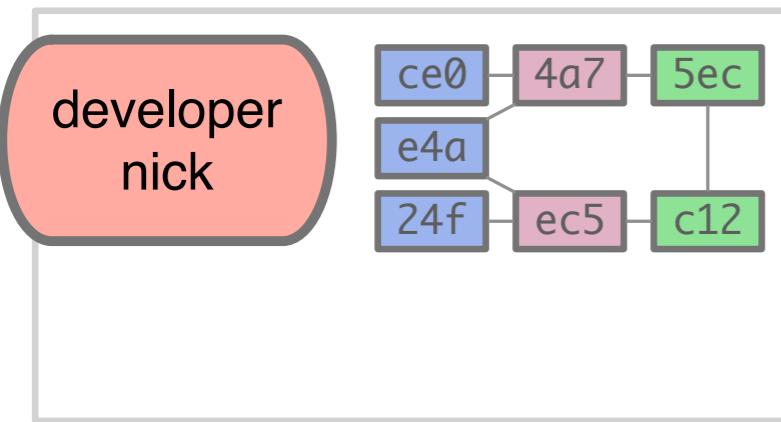


git fetch jess

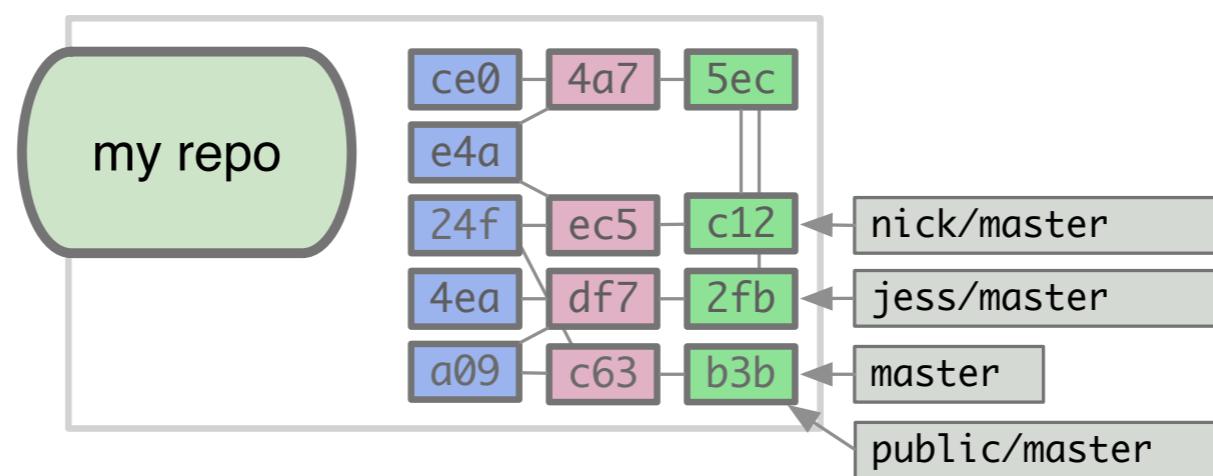
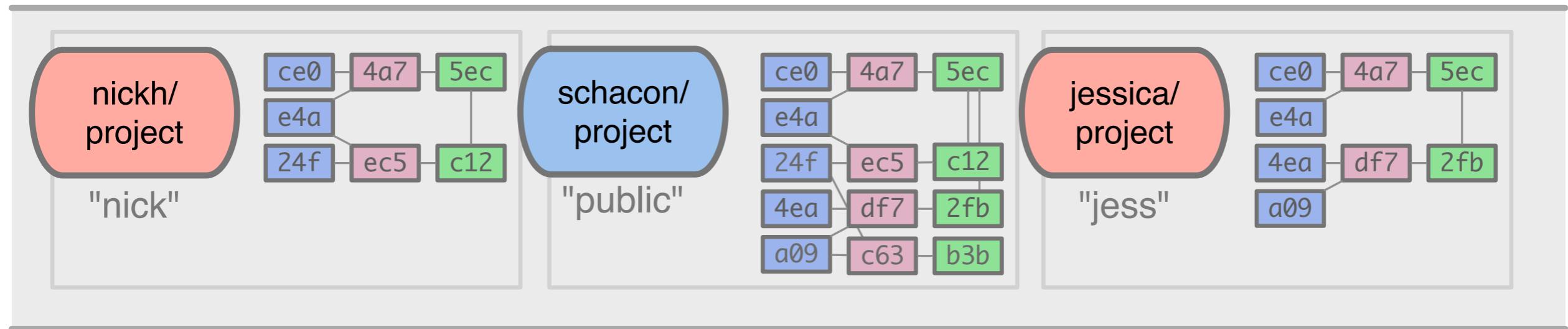
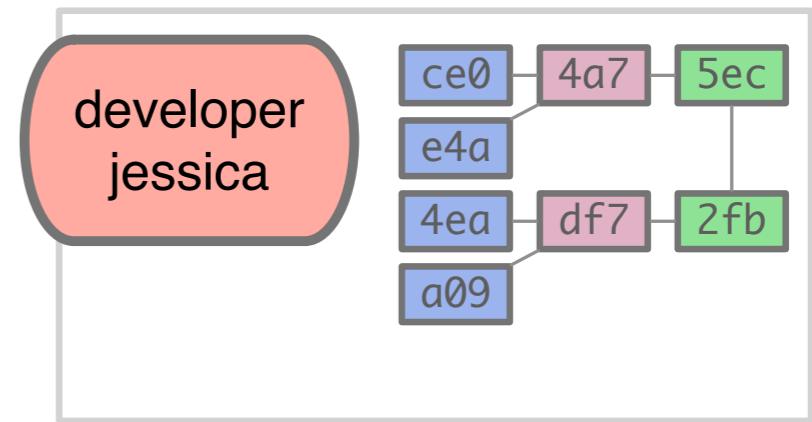
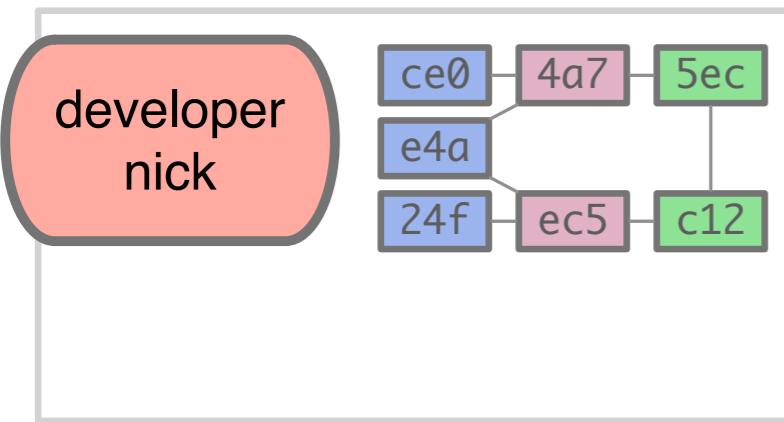


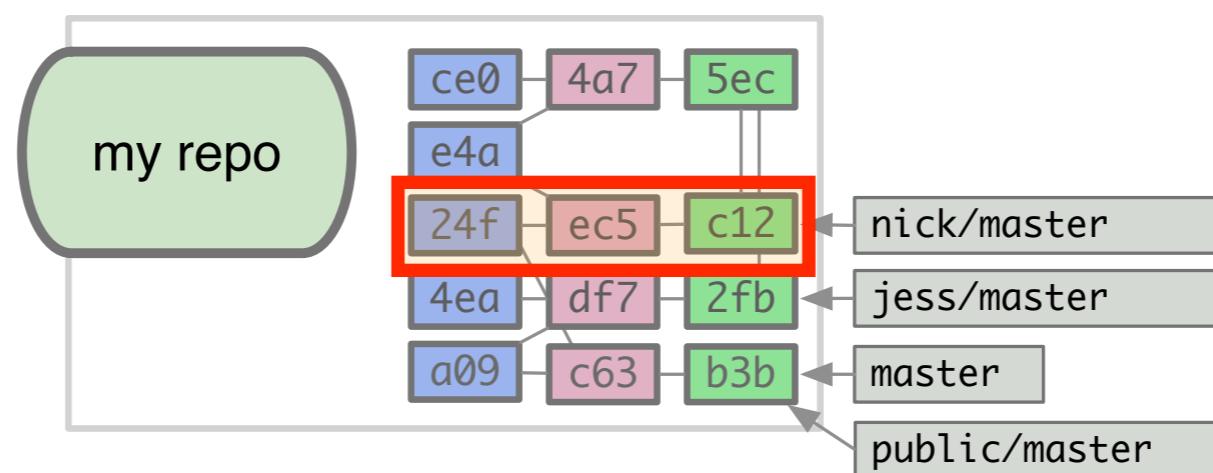
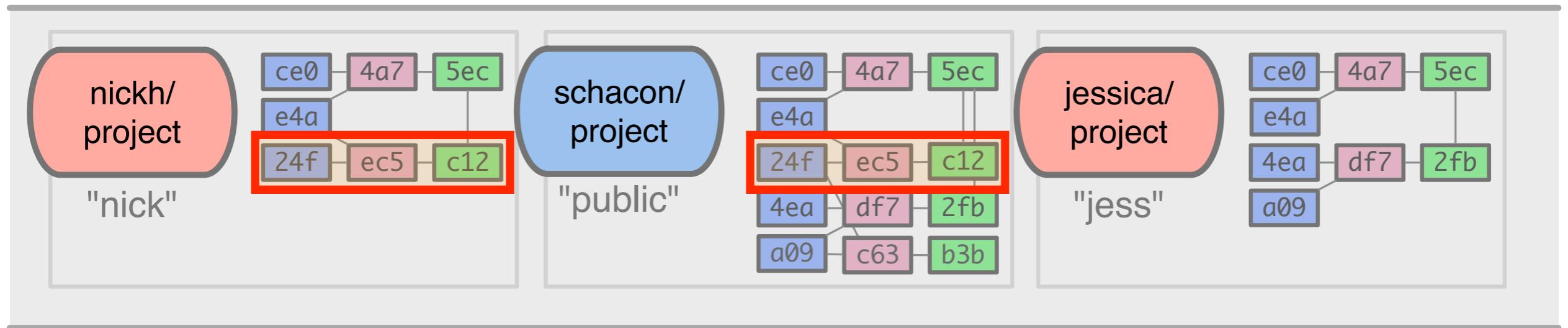
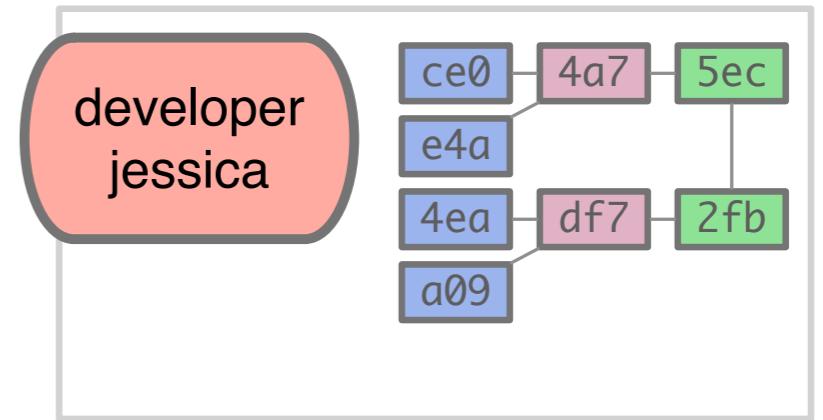
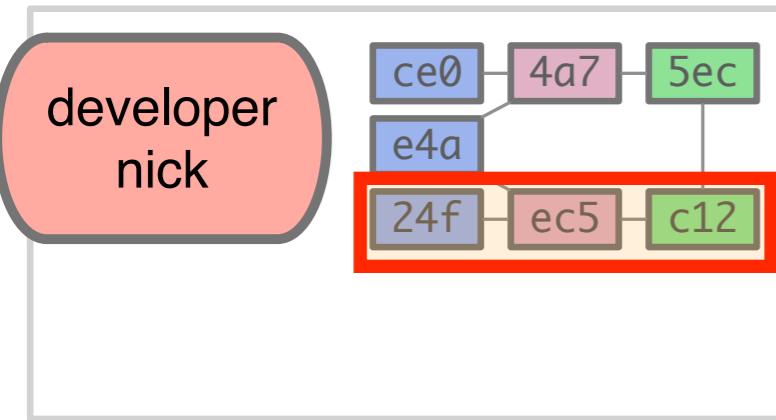


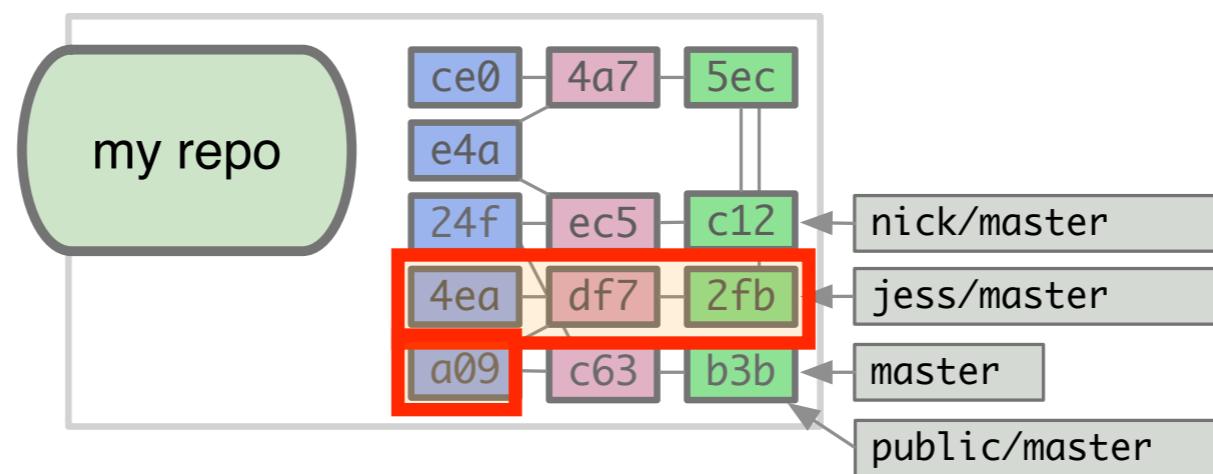
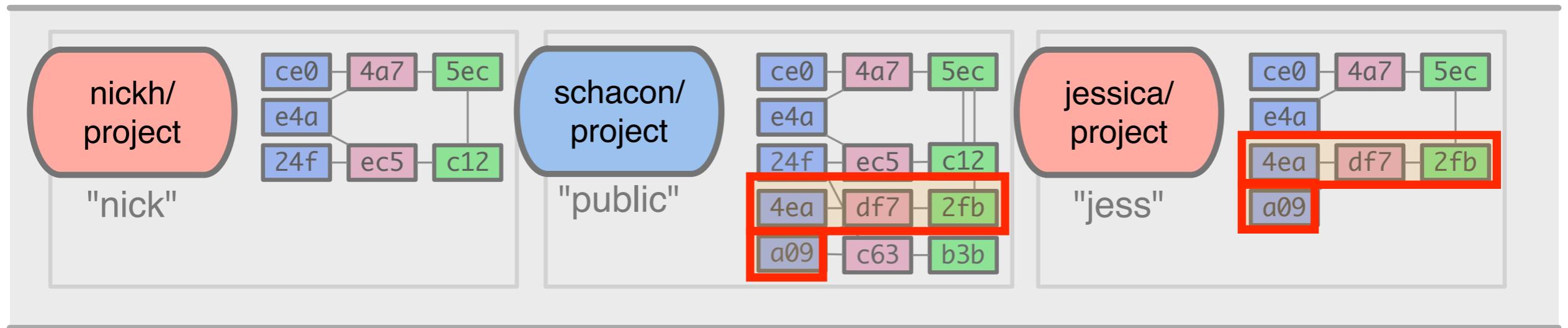
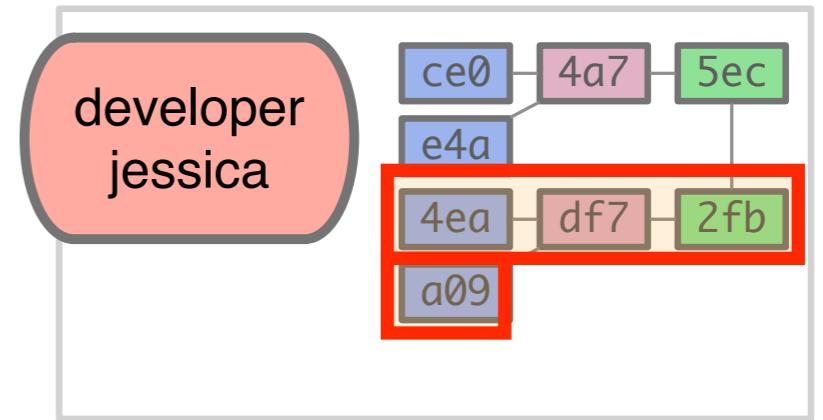
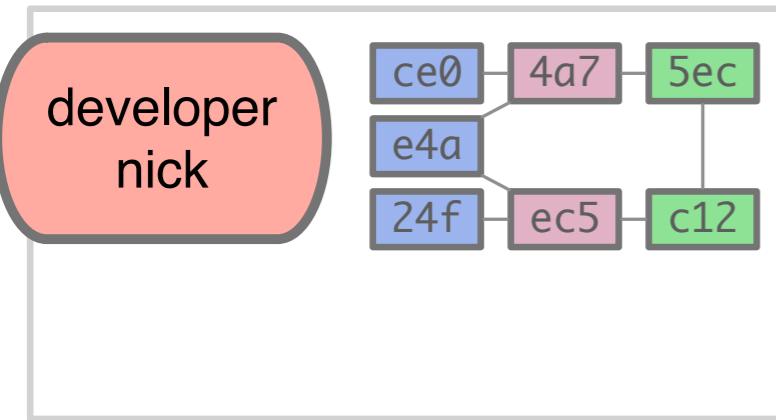
git merge nick jess

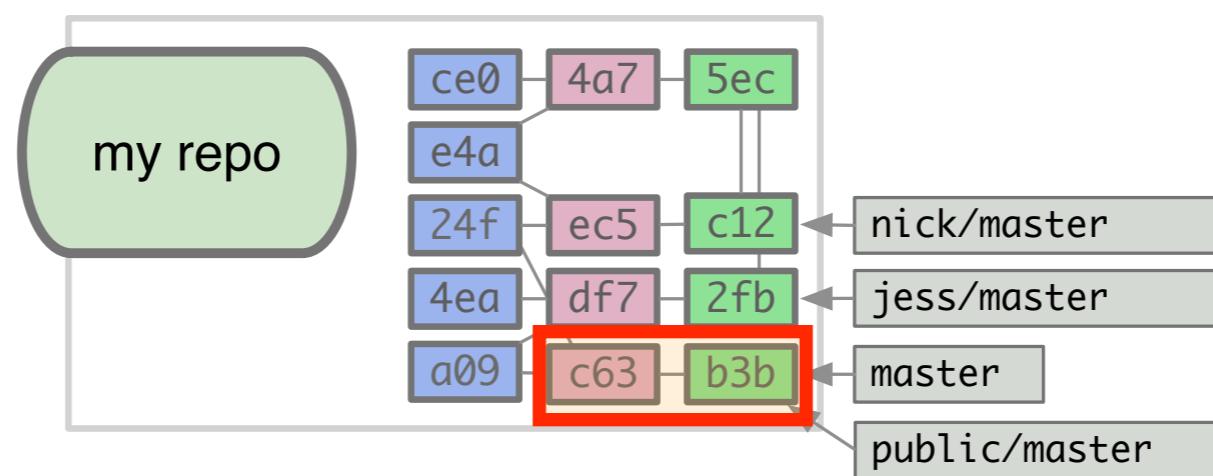
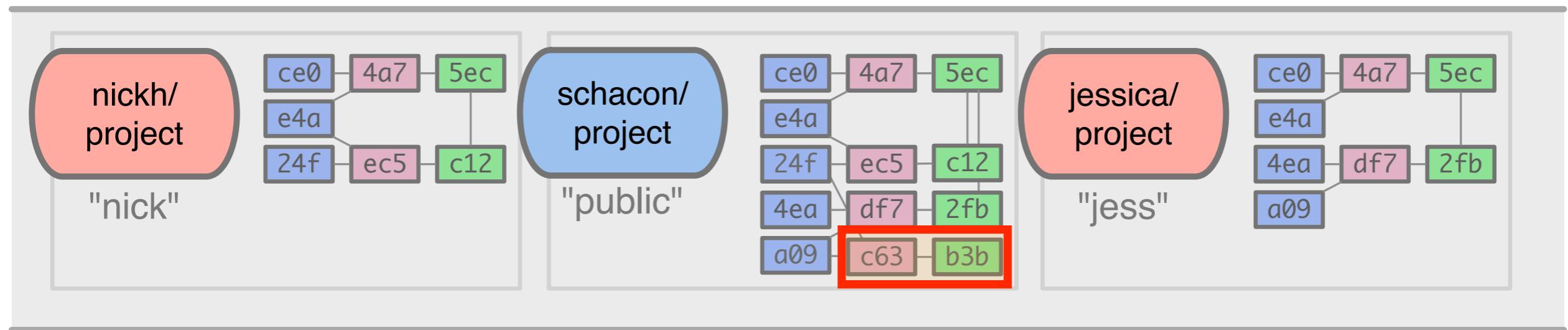
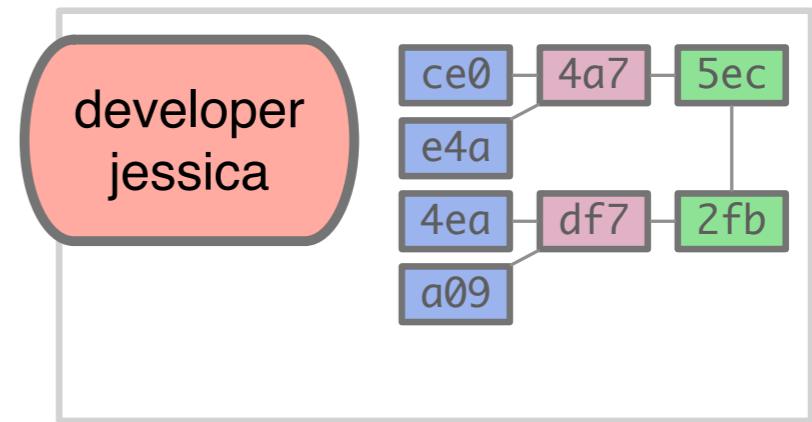
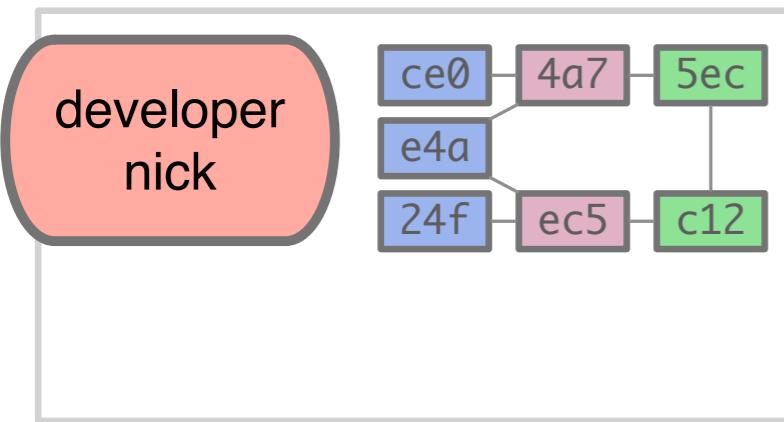


git push public





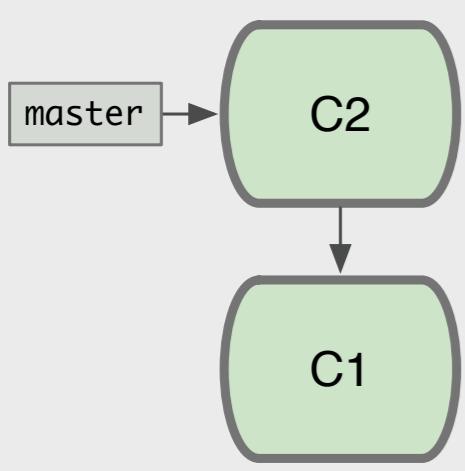


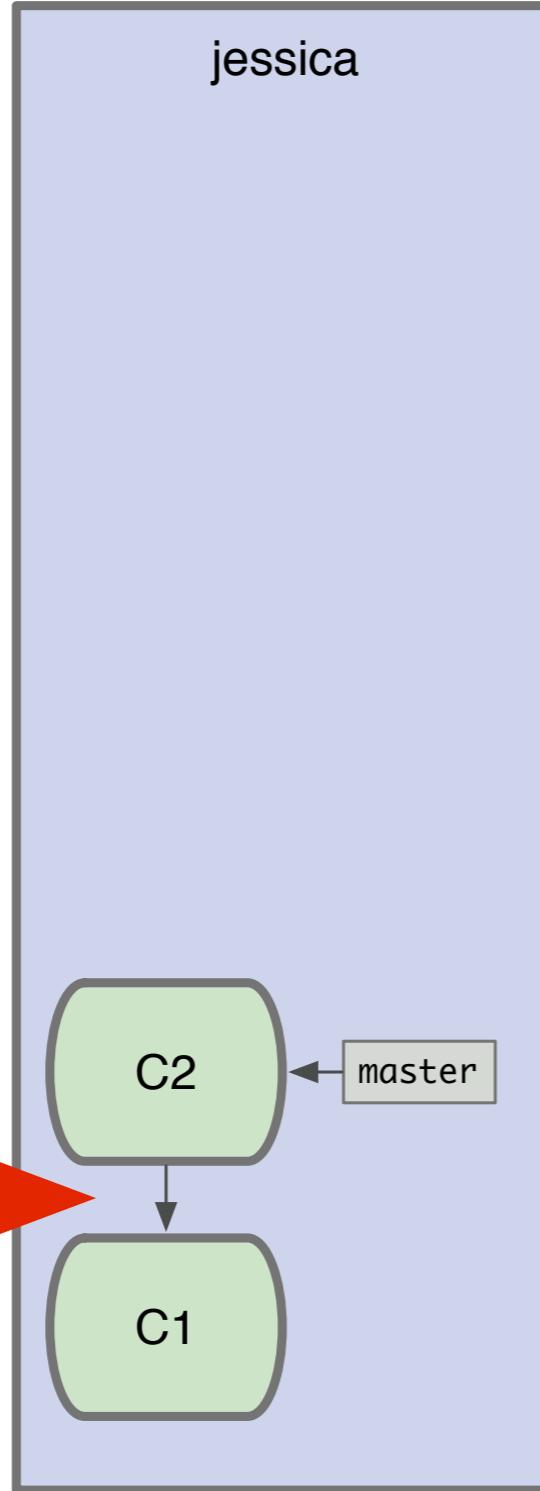


rebasing

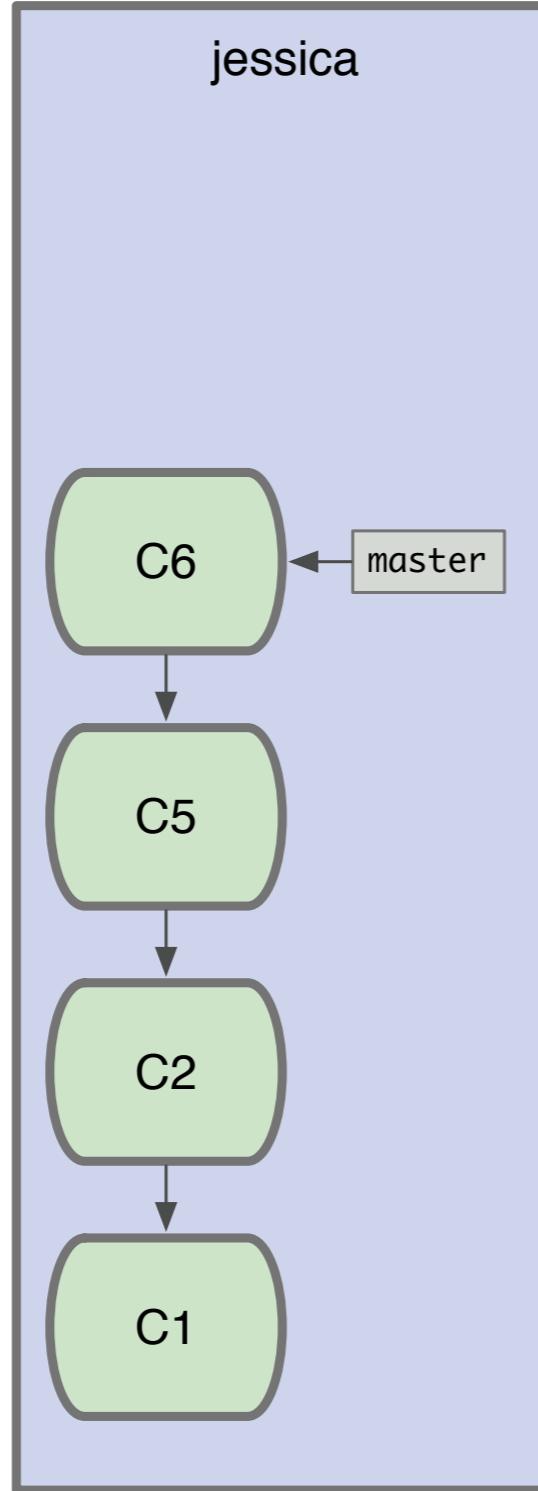
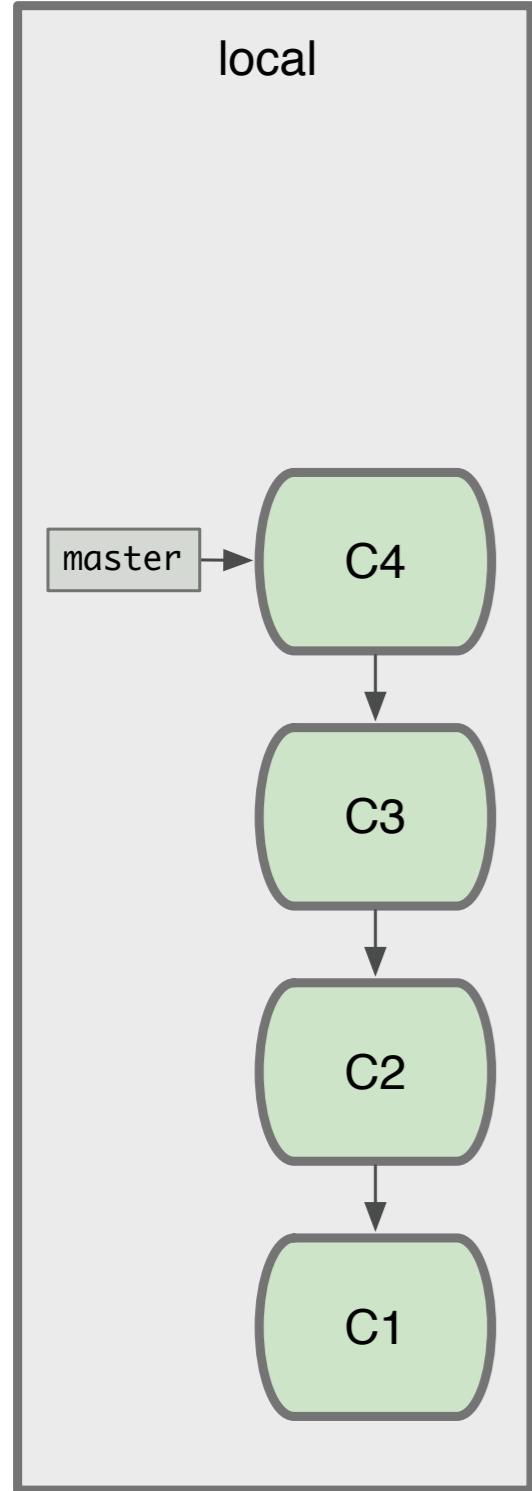
local

jessica

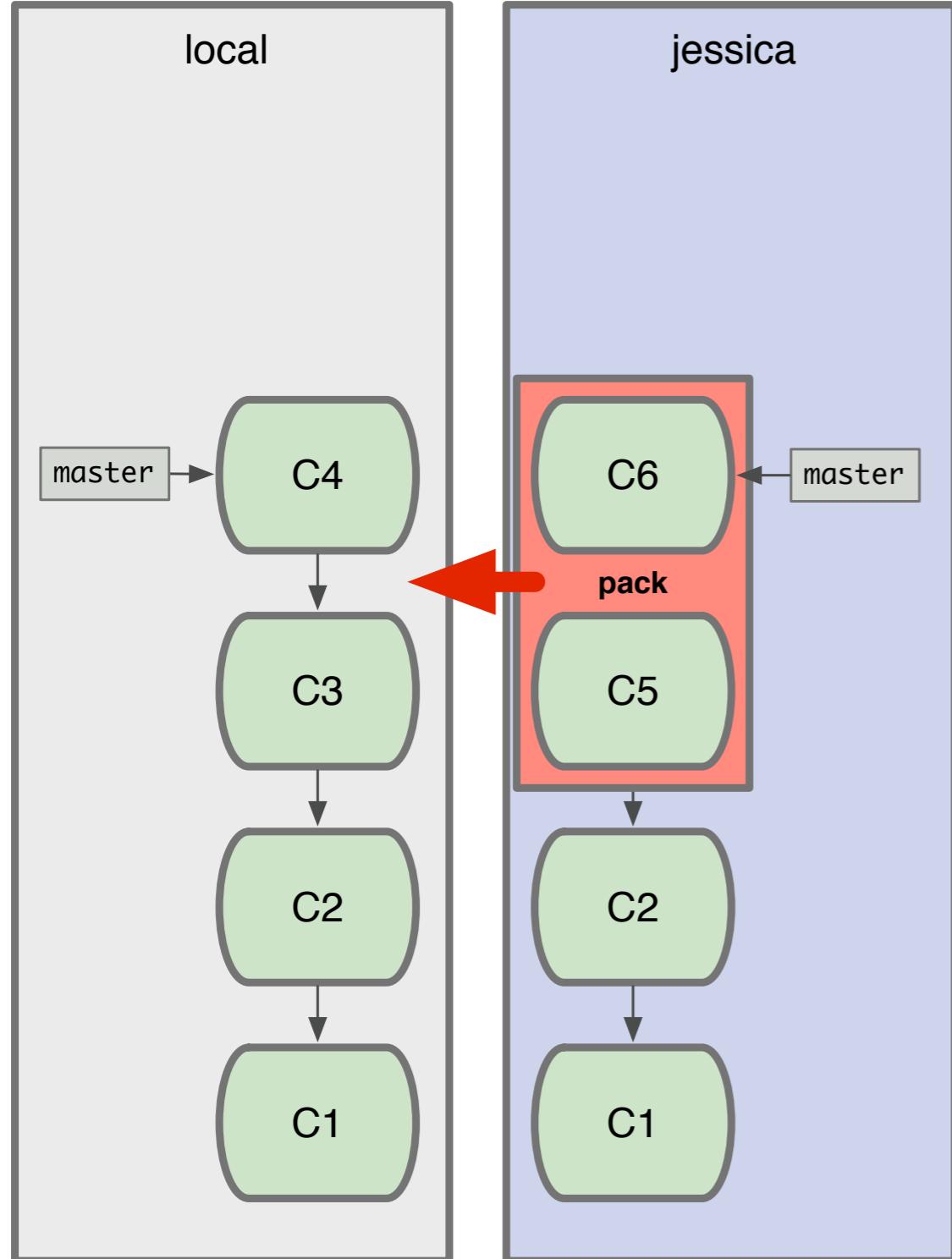




git clone

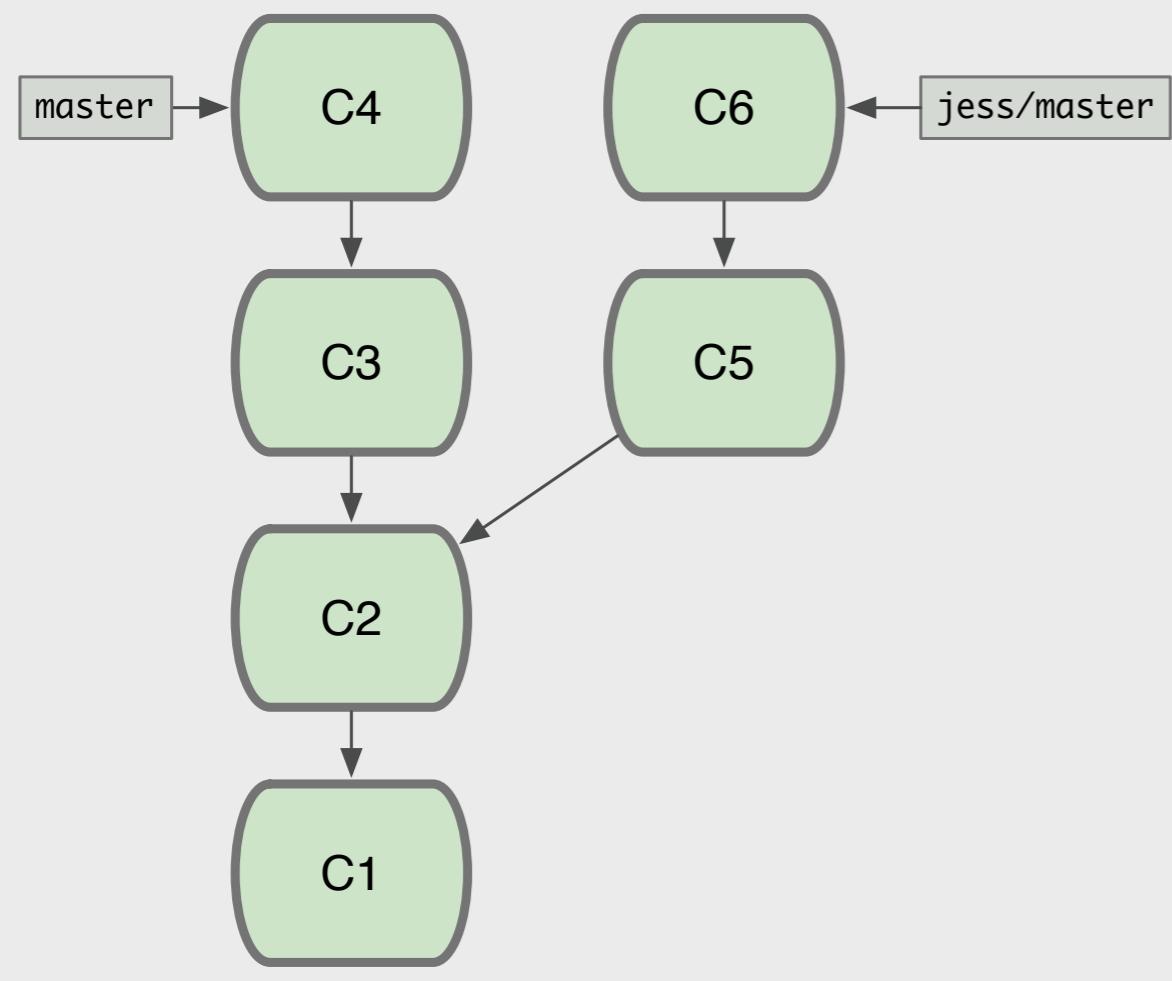


git commit
git commit

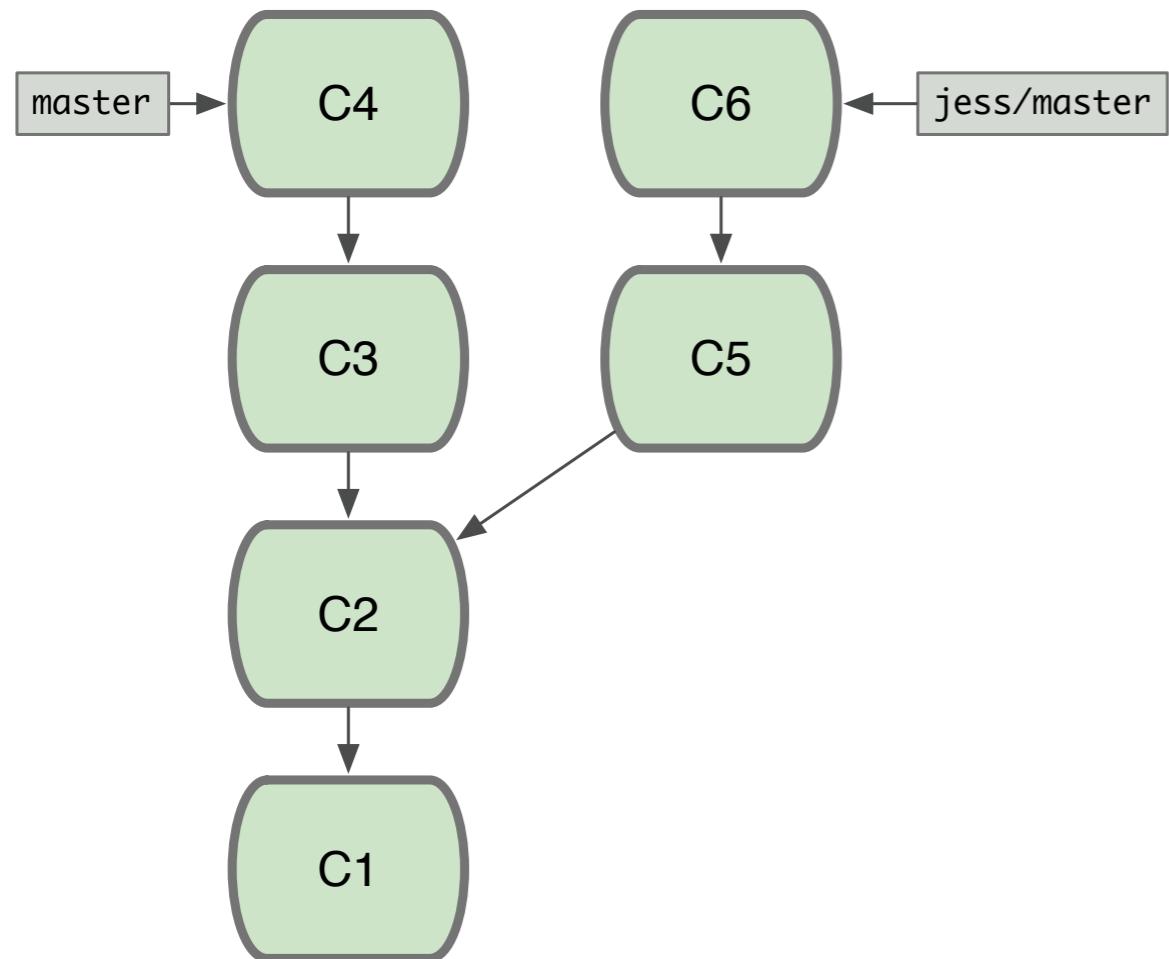


git fetch jess

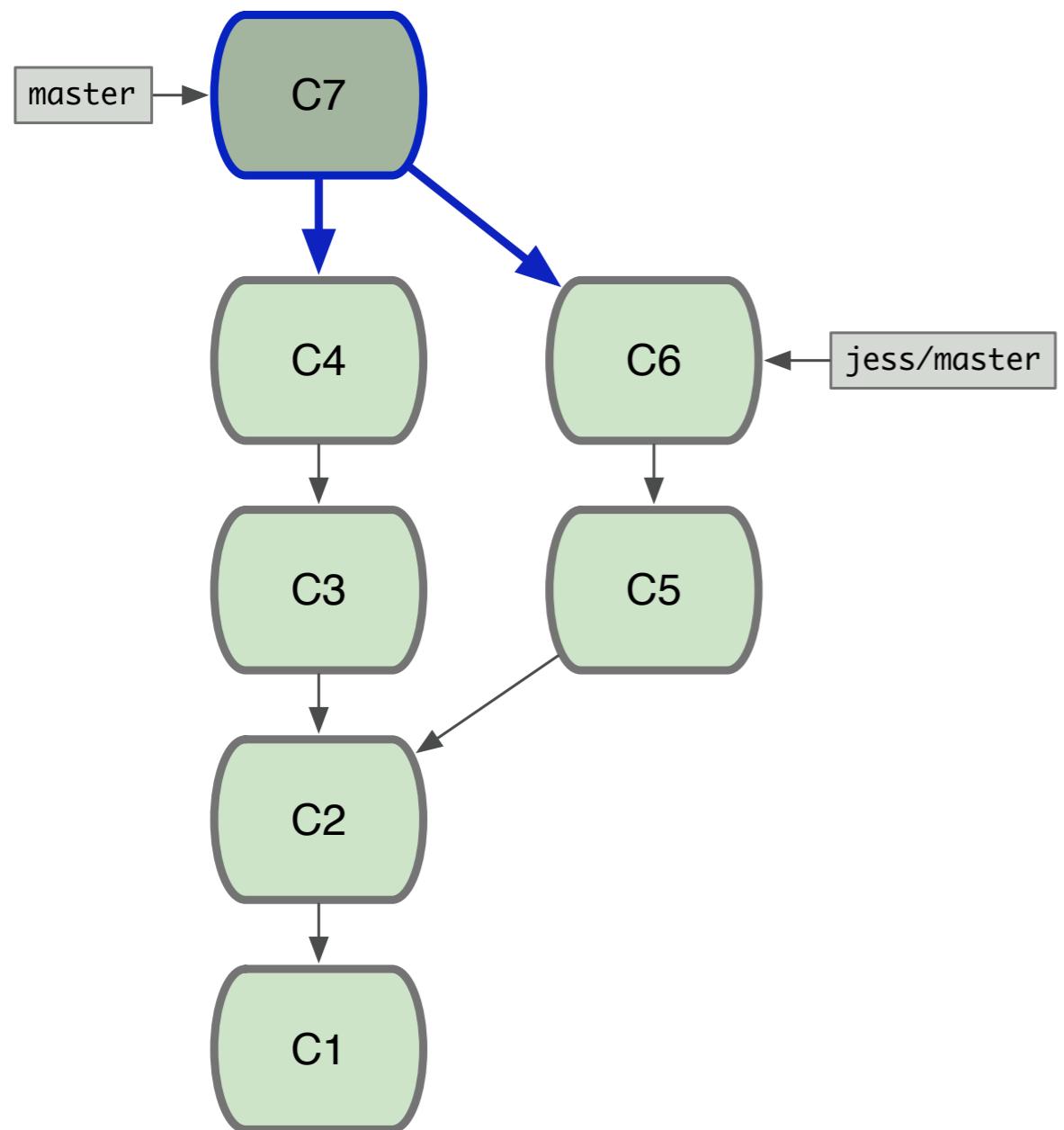
local



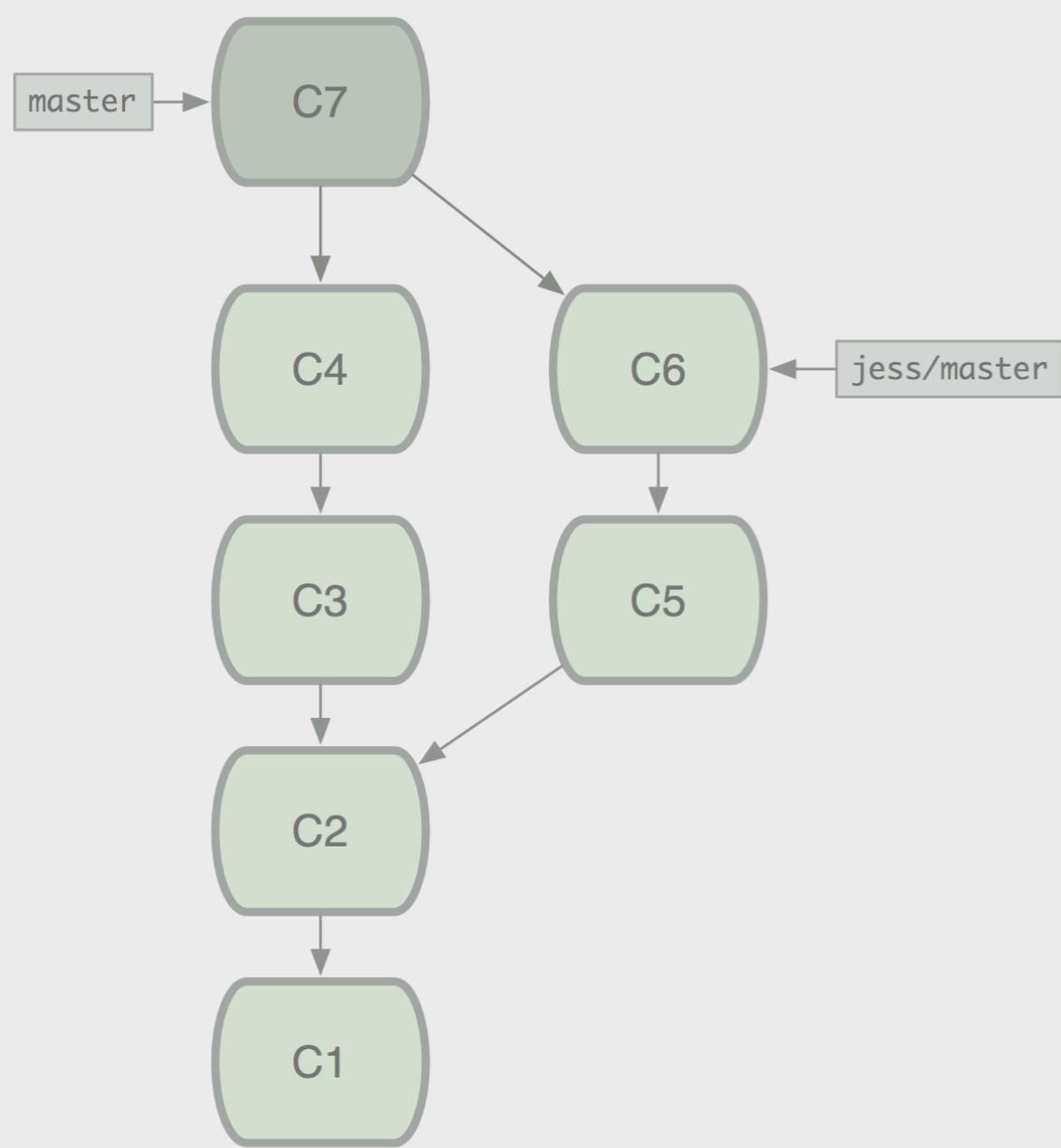
git merge



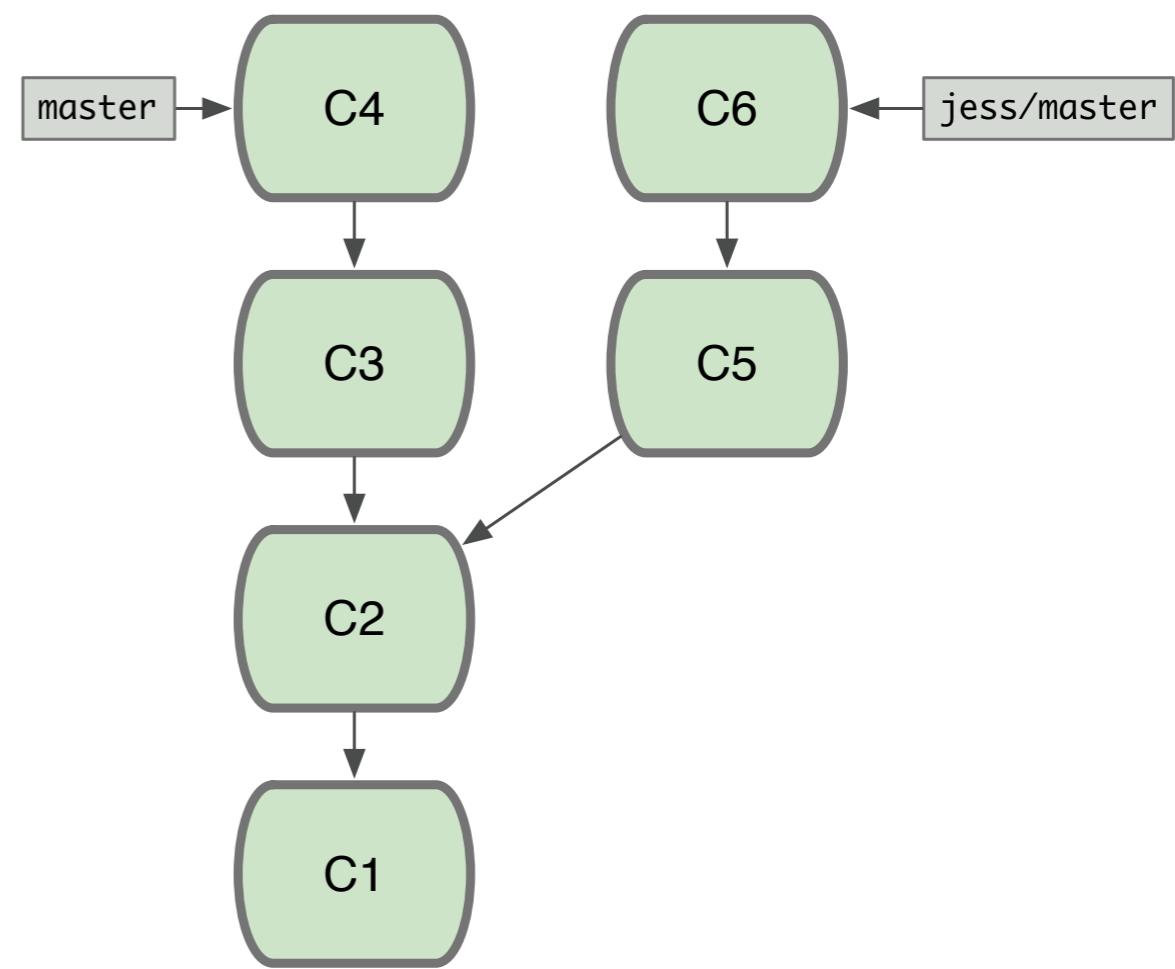
git merge



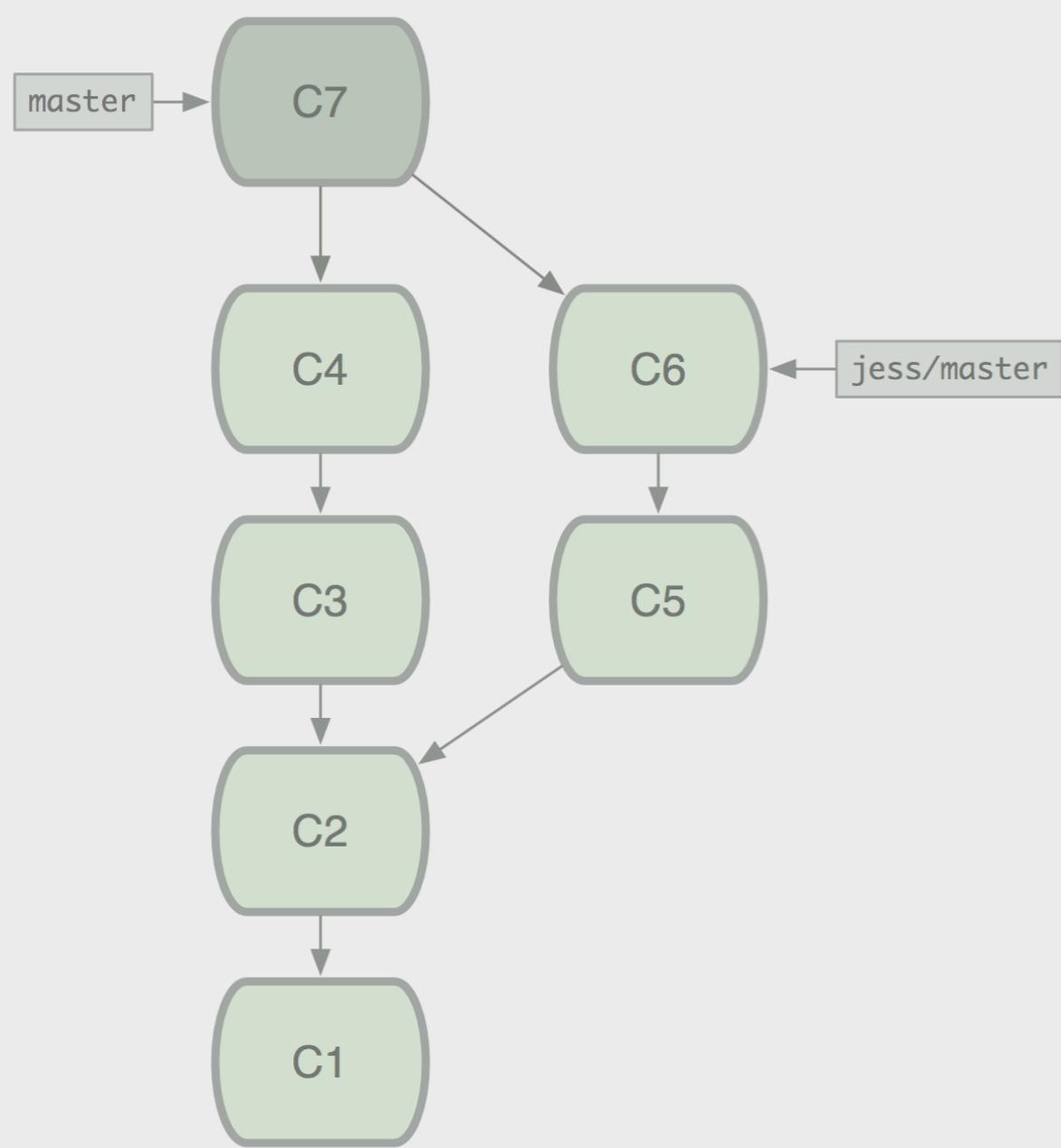
git merge



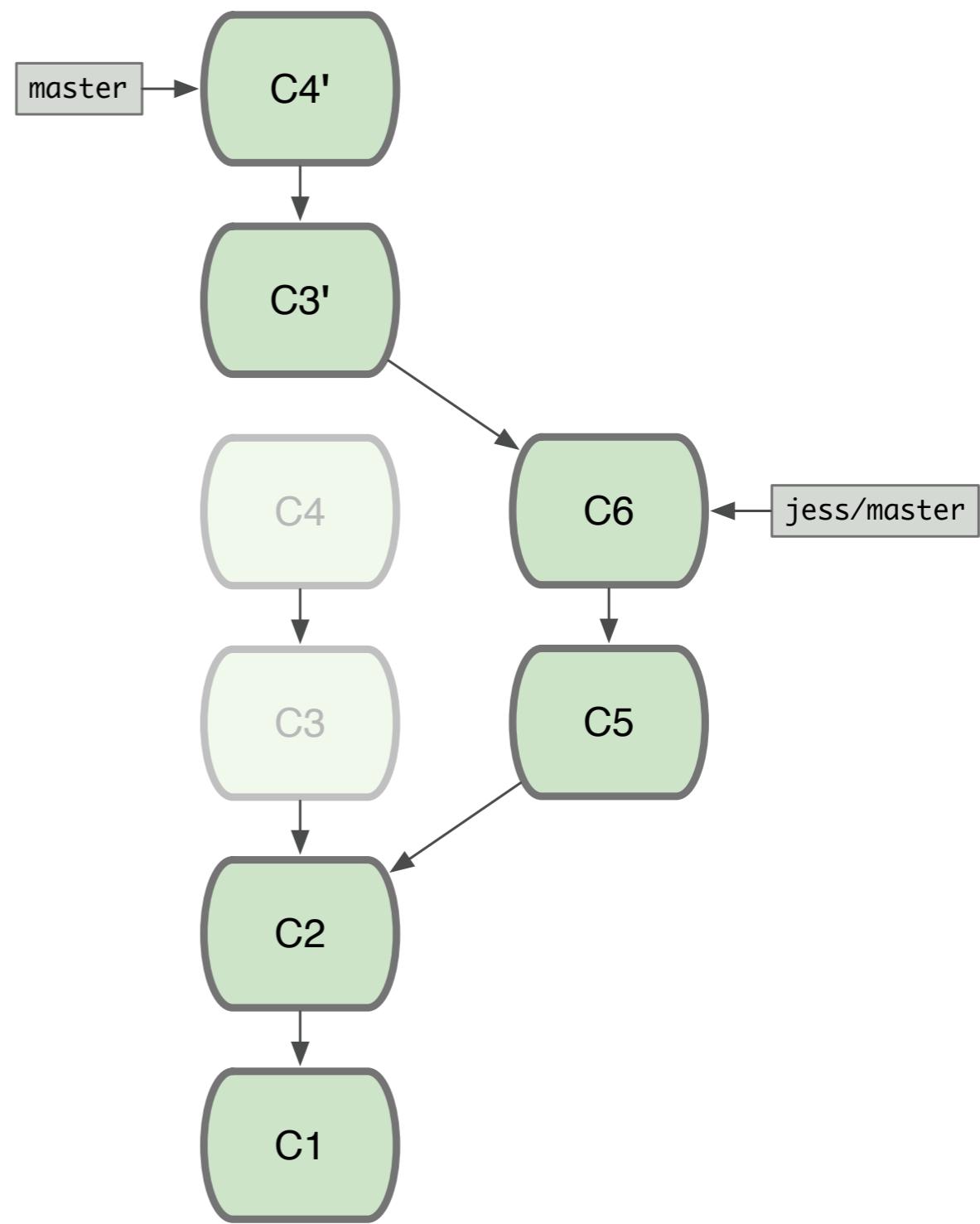
git rebase



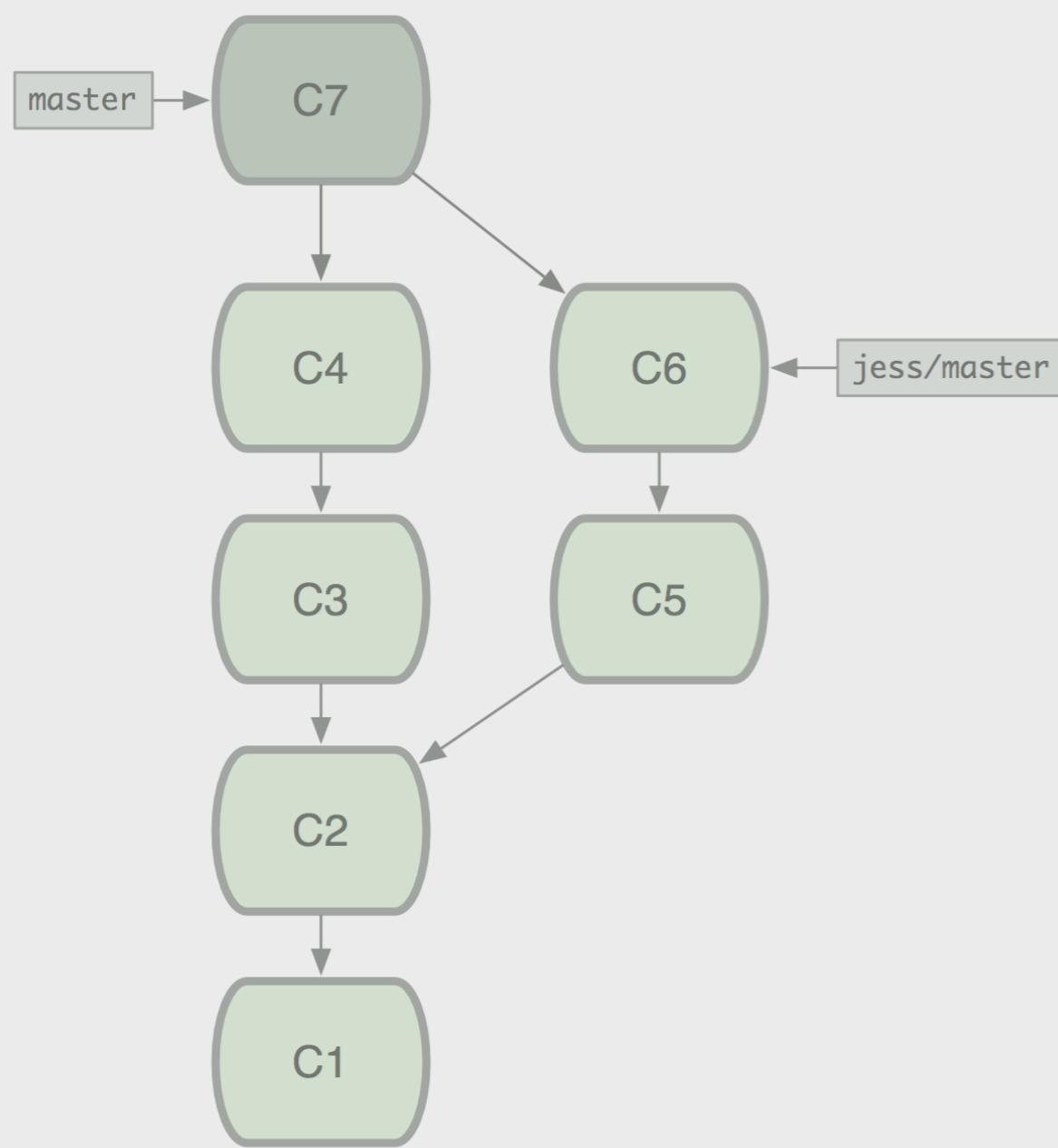
git merge



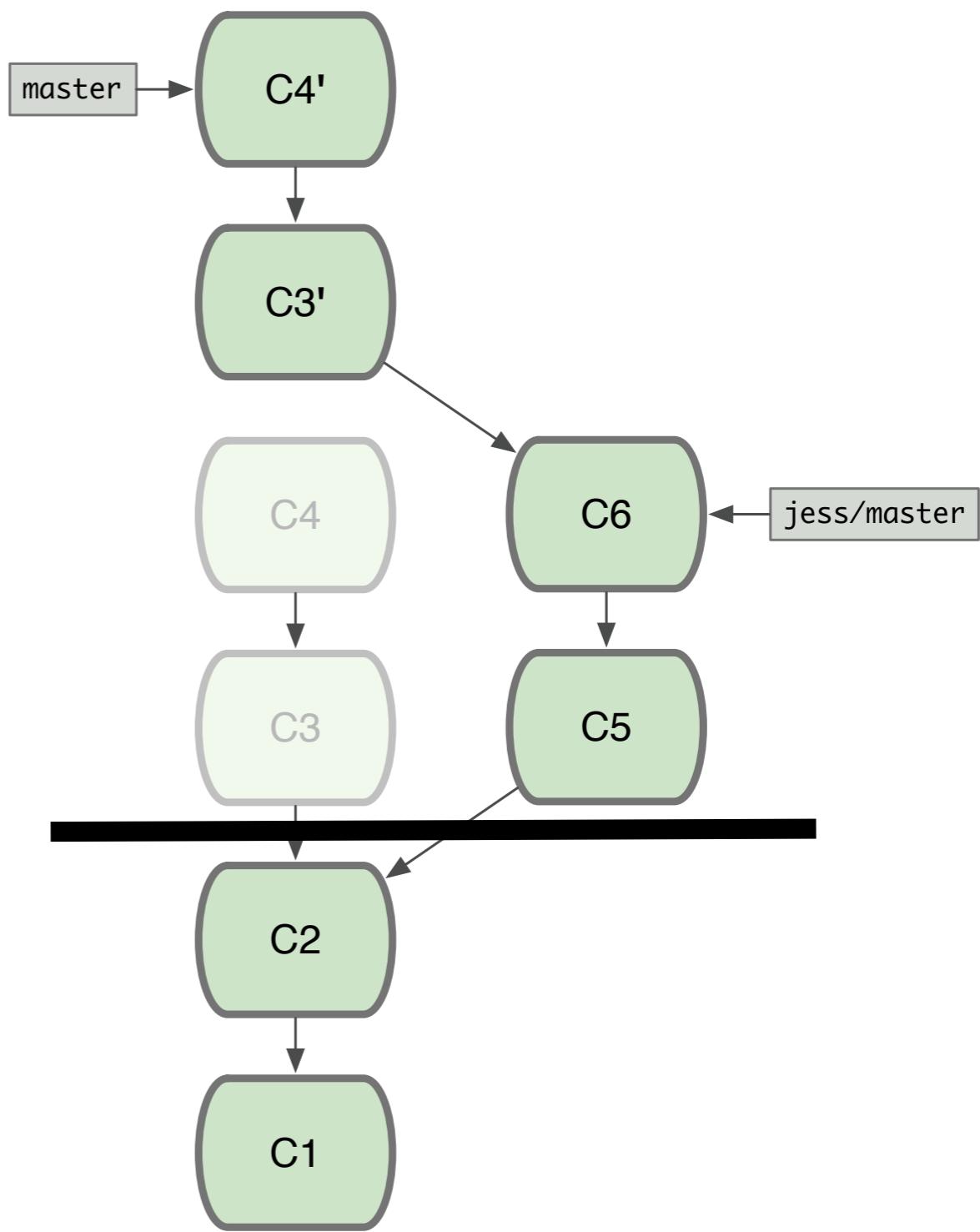
git rebase



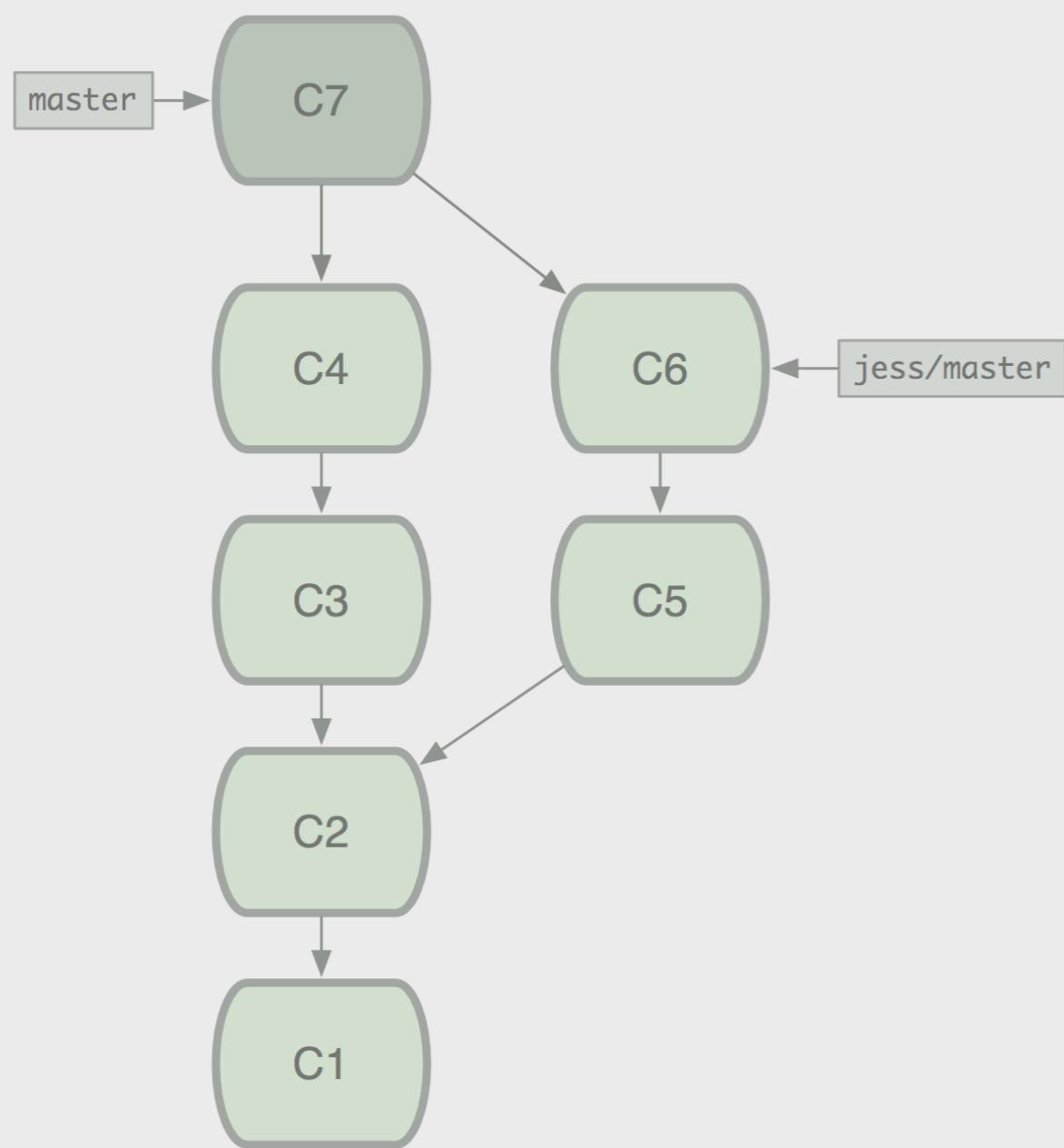
git merge



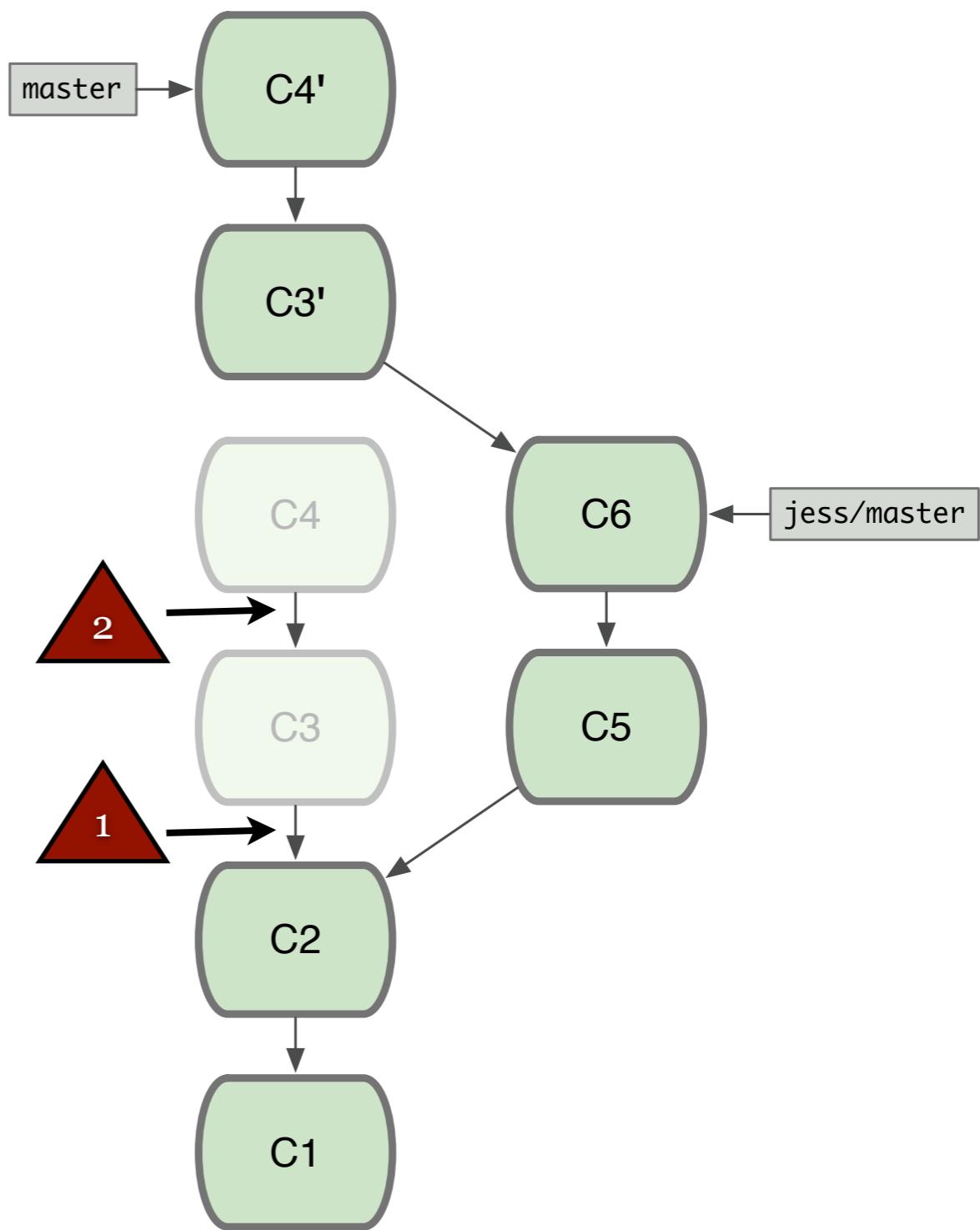
git rebase



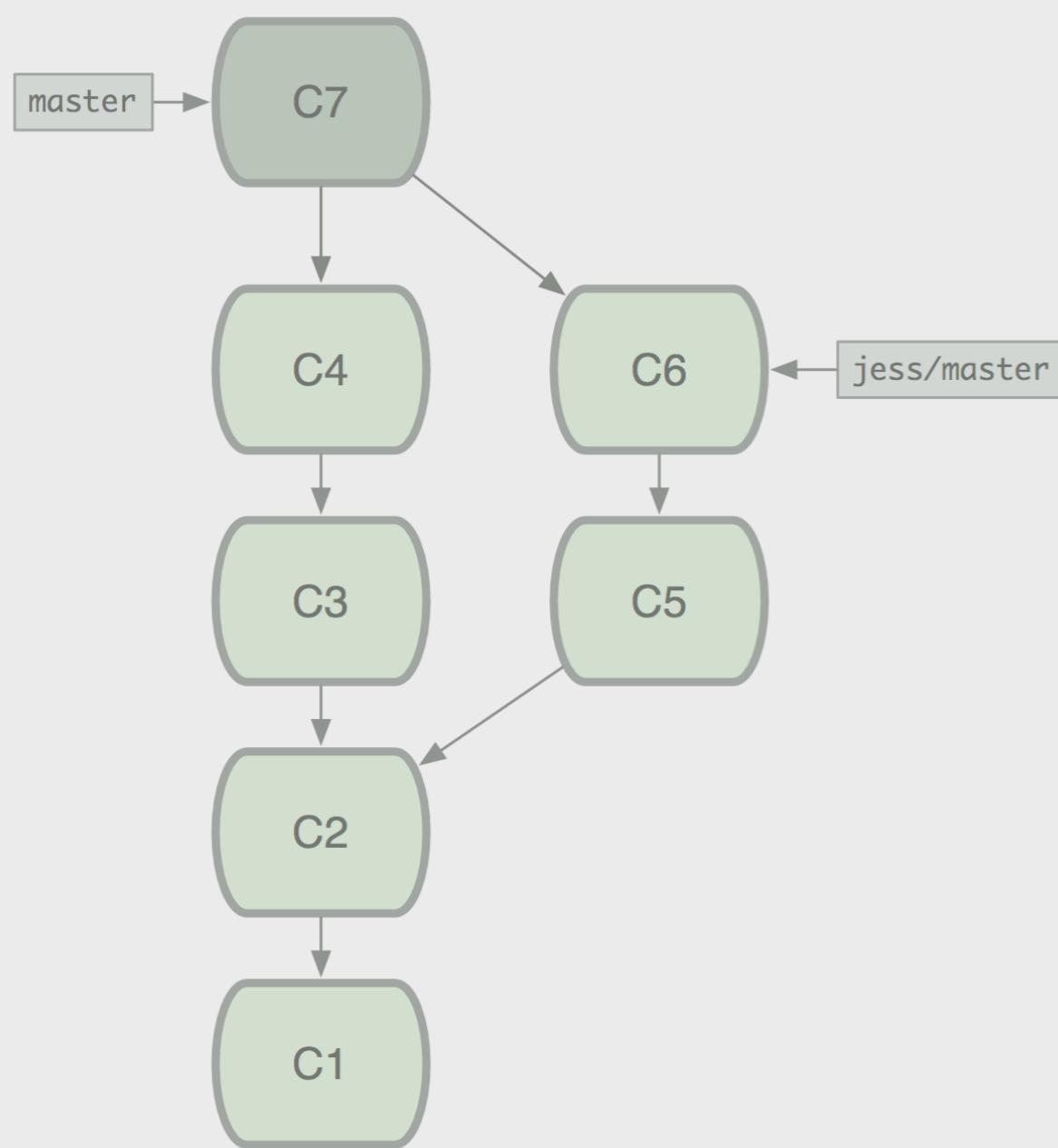
git merge



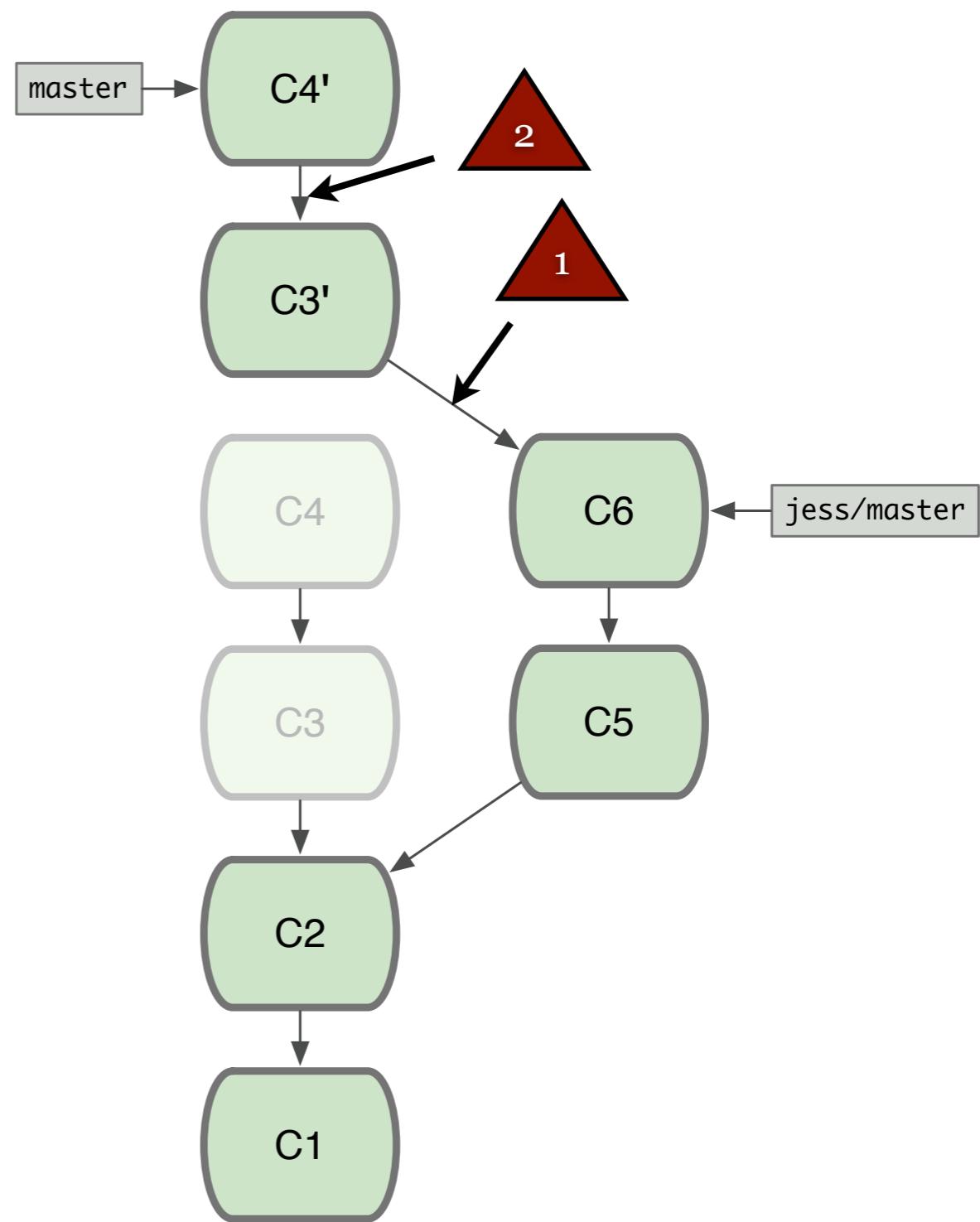
git rebase



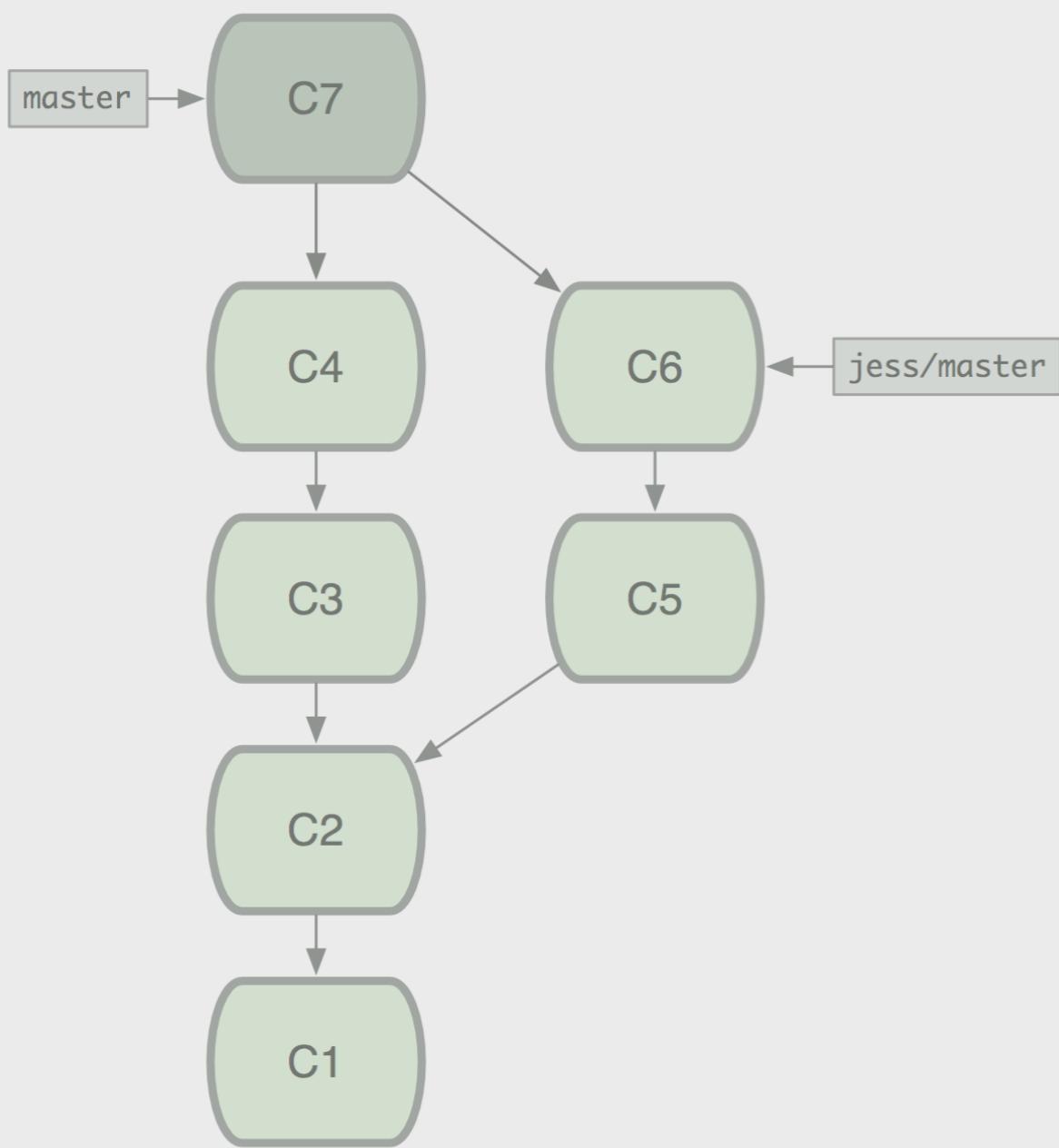
git merge



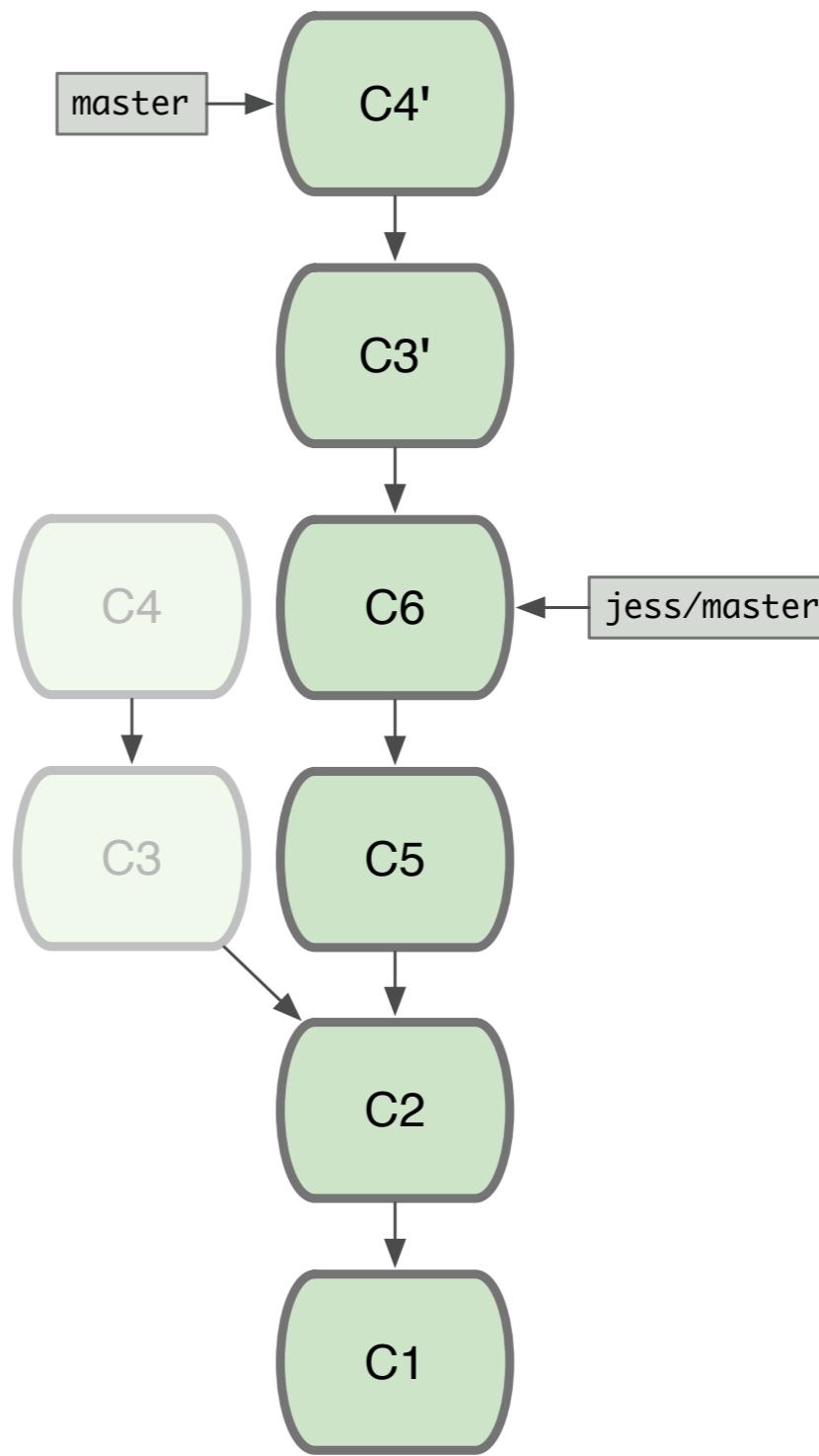
git rebase



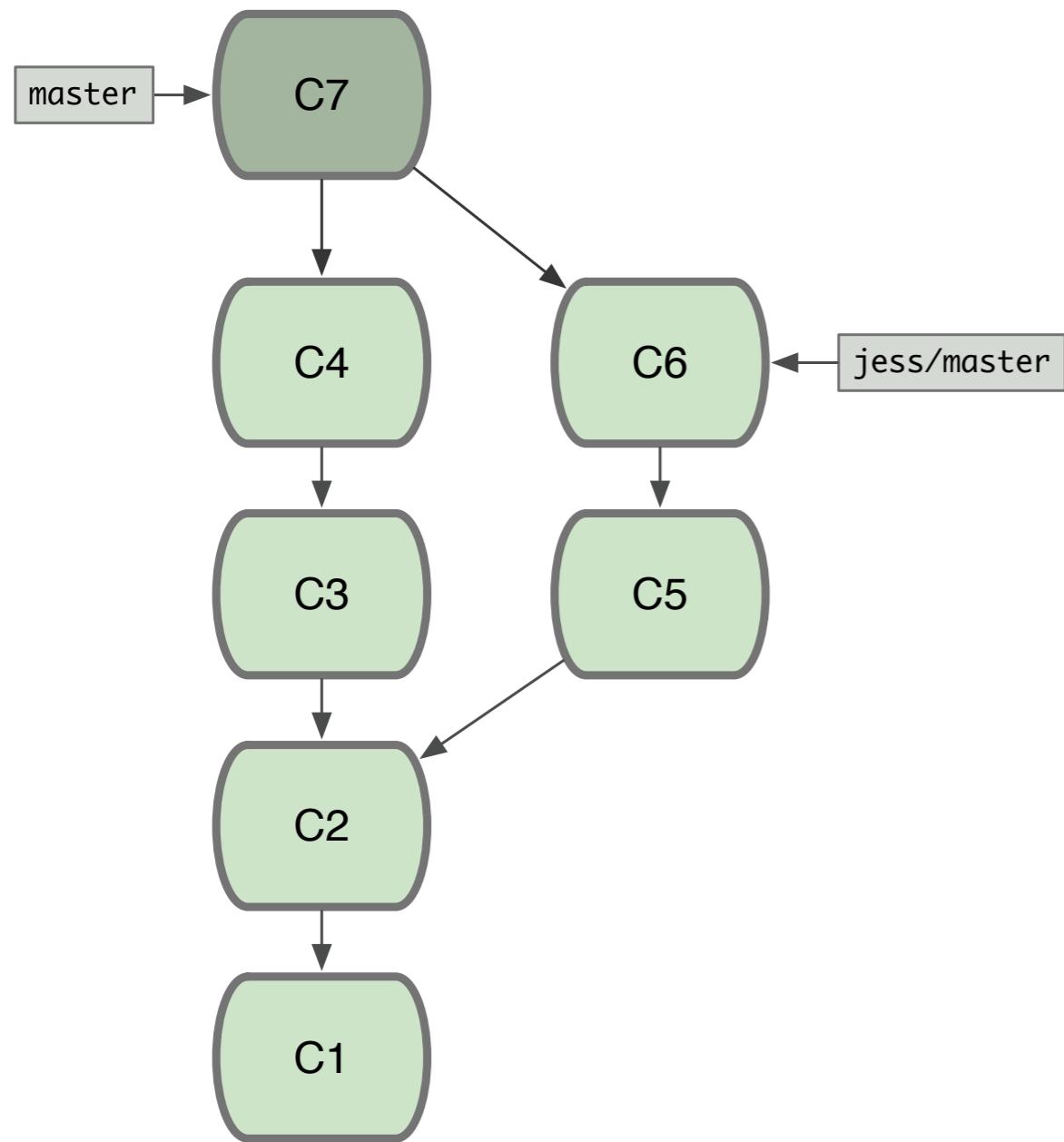
git merge



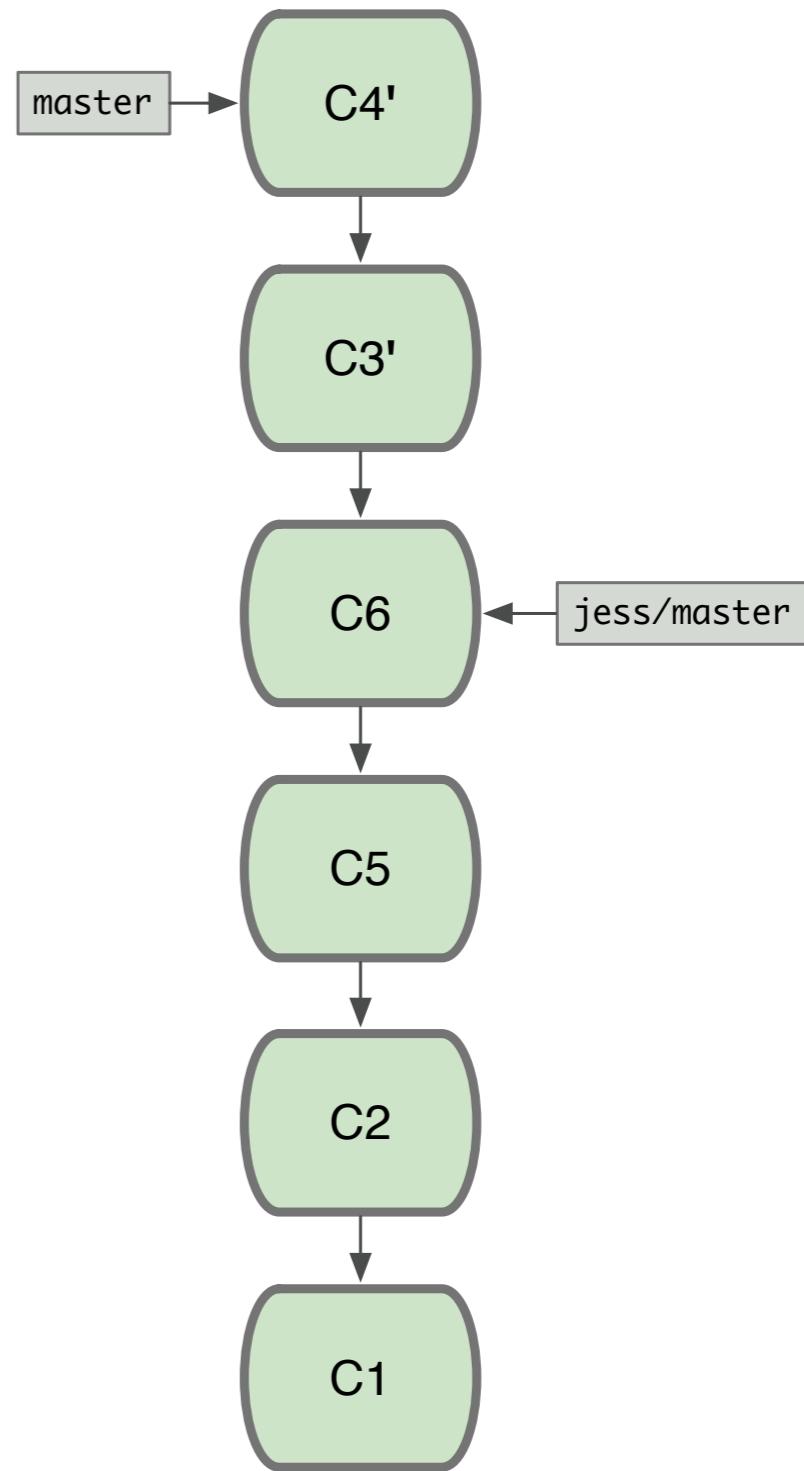
git rebase



git merge



git rebase



the treeish

the treeish

alternate ways to refer to
objects or ranges of objects

Treeish

- full sha-1
- partial sha-1
- branch or tag name
- date spec
- ordinal spec
- carrot parent
- tilde spec
- tree pointer
- blob spec
- ranges

Full SHA1

6e453f523fa1da50ecb04431101112b3611c6a4d

Partial SHA1

6e453f523fa1da50ecb04431101112b3611c6a4d

6e453f523fa1da50

6e453

Branch, Remote or Tag Name

v1.0

master

origin/testing

Date Spec

master@{yesterday}

master@{1 month ago}

Ordinal Spec

master@{5}

5th prior value of ‘master’

Carrot Parent

master^{^2}

2nd parent of ‘master’

Tilde Spec

master~2

2nd generation grandparent of ‘master’

Tree Pointer

`master^{tree}`

tree that ‘master’ points to

Blob Spec

master:/path/to/file

blob of that file in ‘master’ commit

Ranges

ce0e4 .. e4272

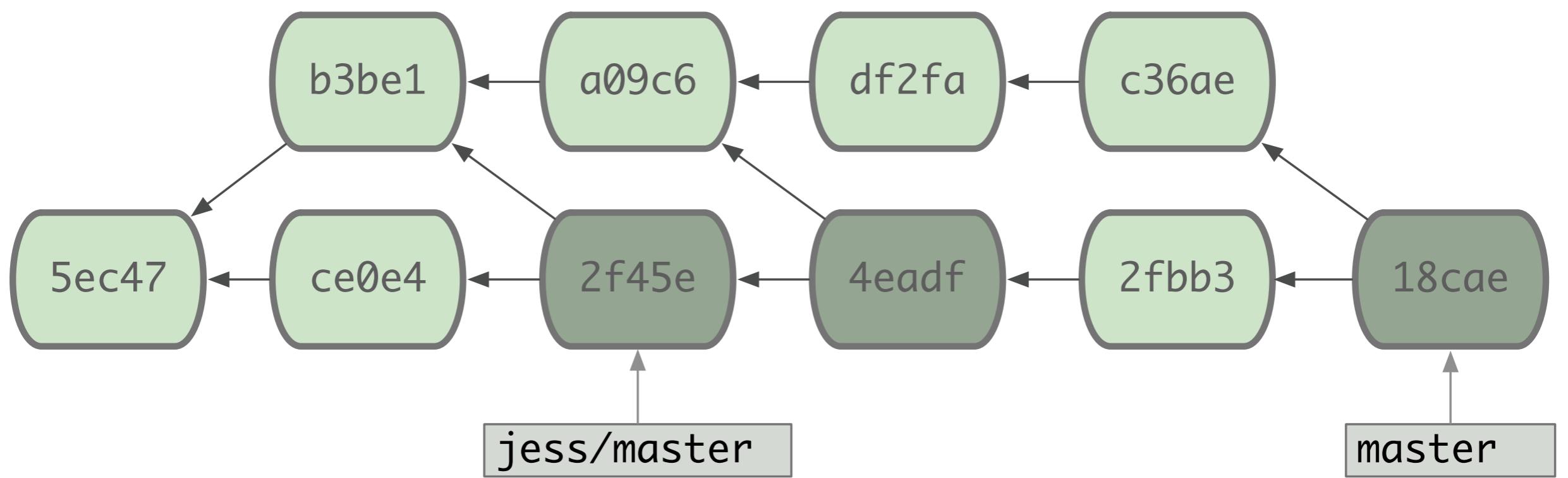
everything between two commits

Ranges

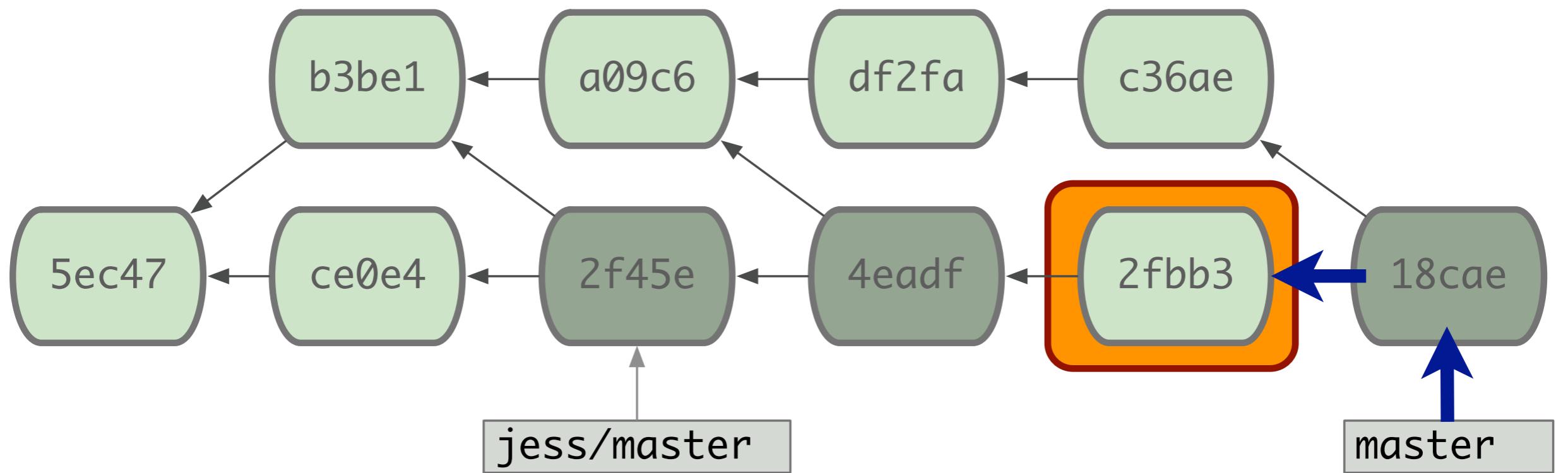
ce0e4 ..

everything since a commit

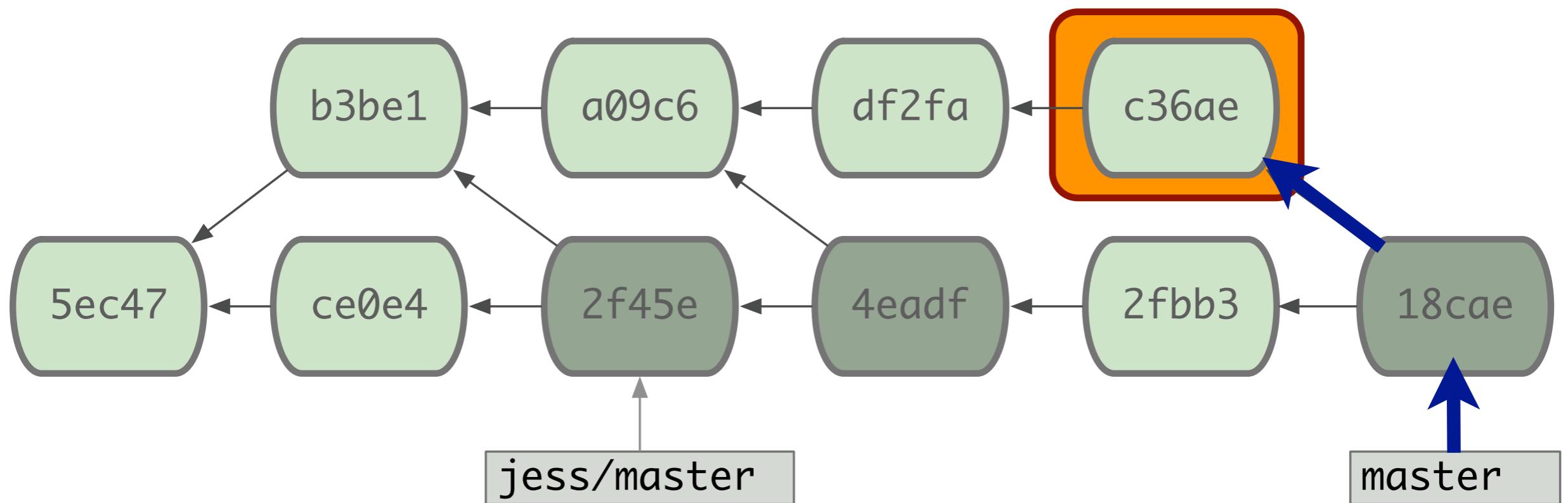
examples



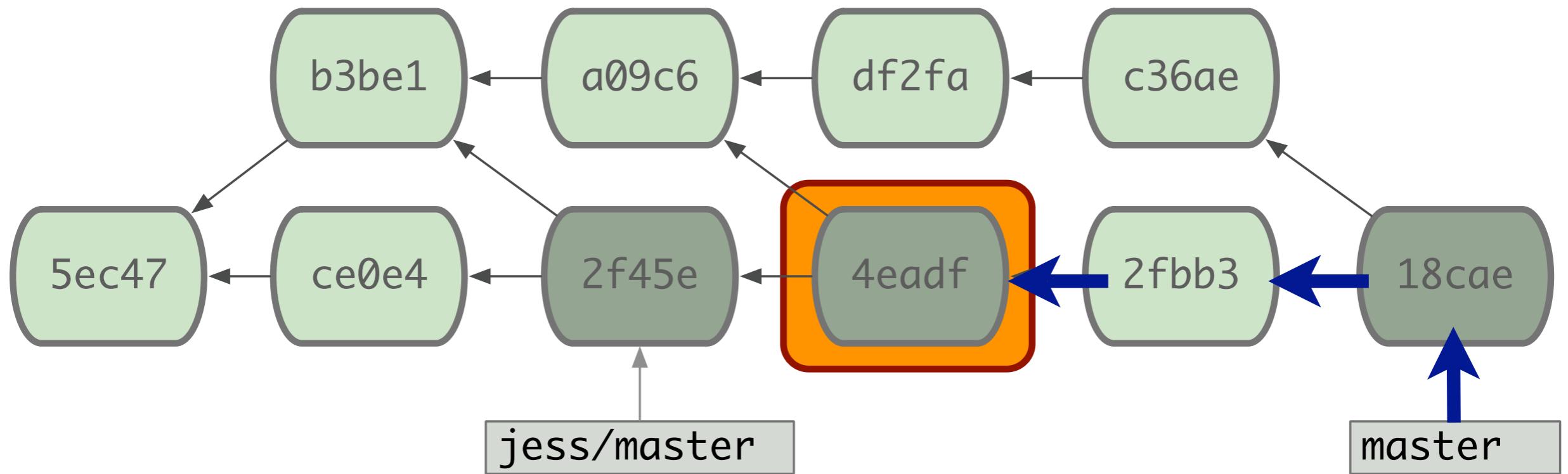
master^Λ



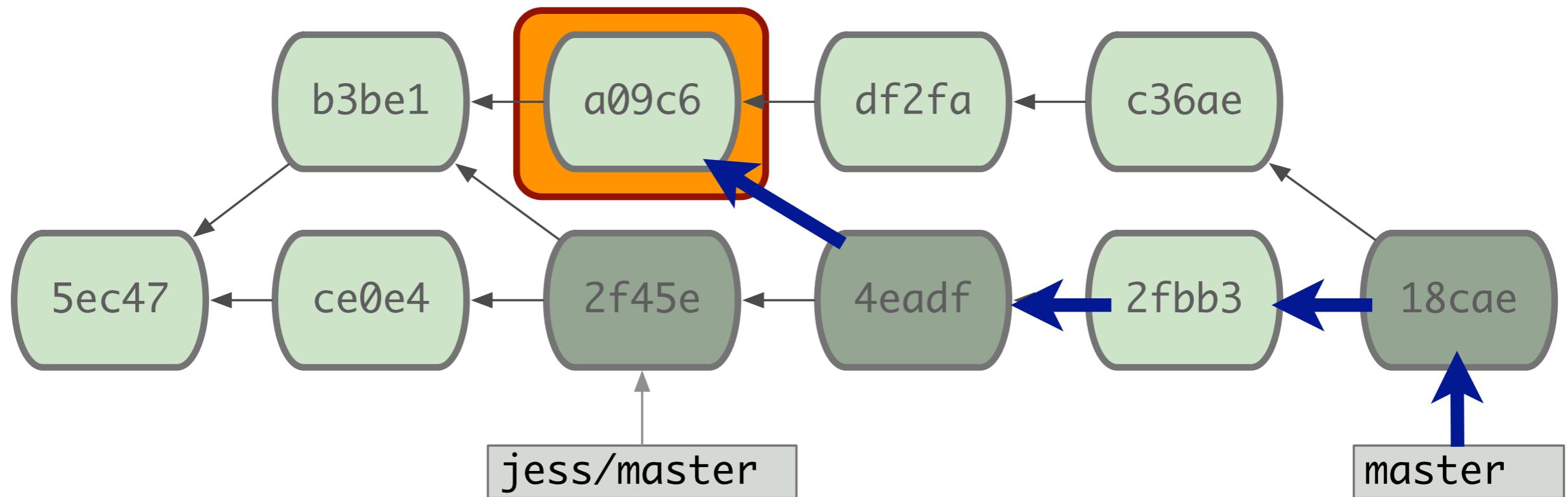
master^{^2}



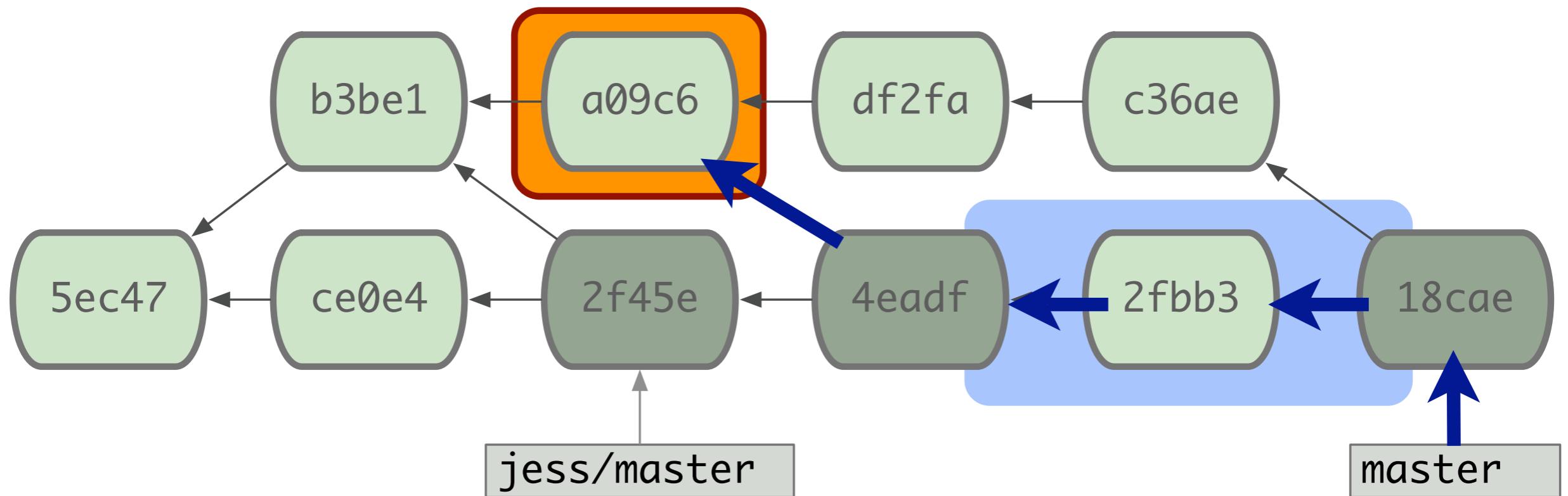
master~2



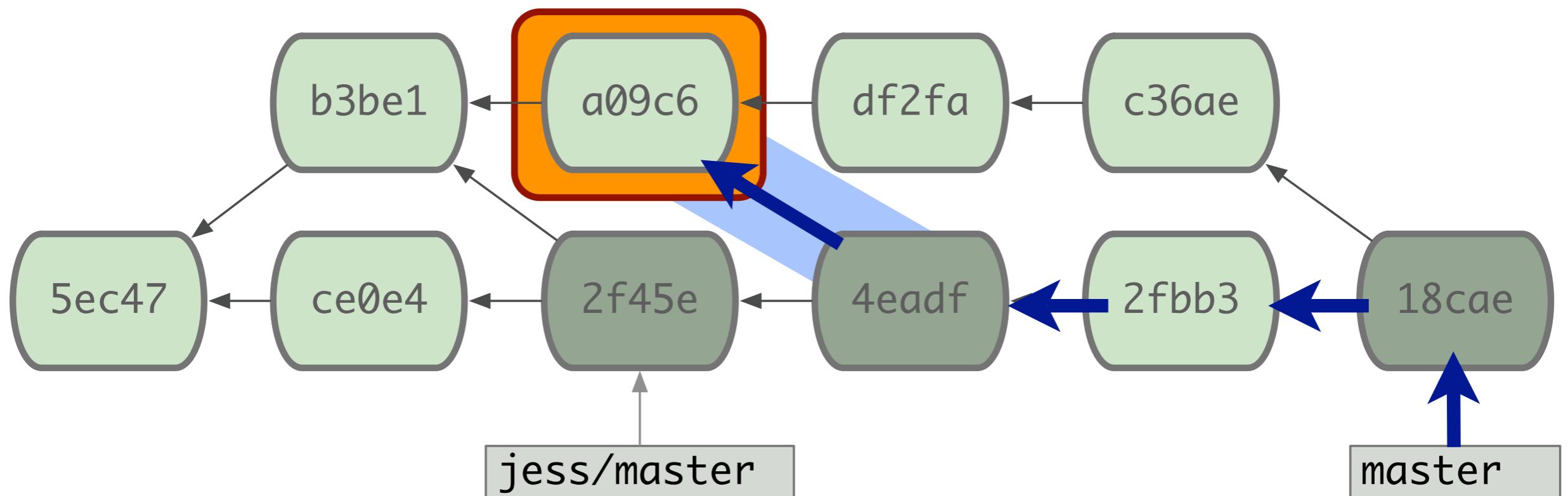
master^{^2} master~2^2



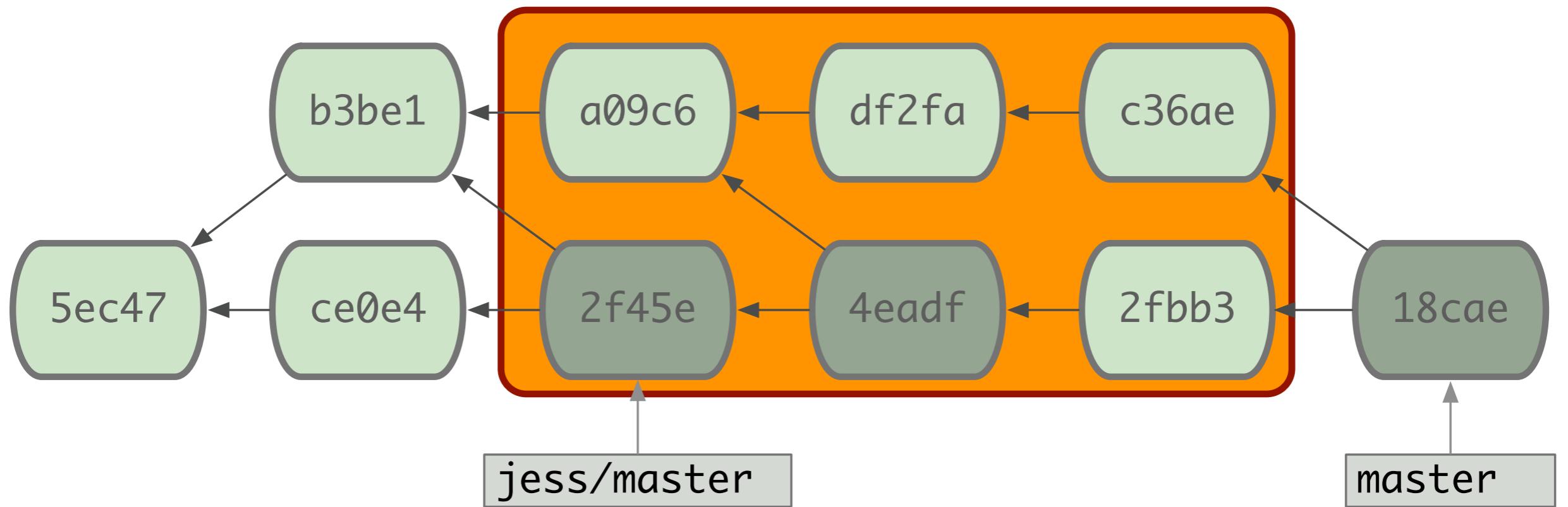
master^{^2}
master~2^2



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



master~3..master^



local stuff

the index

Git Directory

config file

hooks

index

object database

references

index

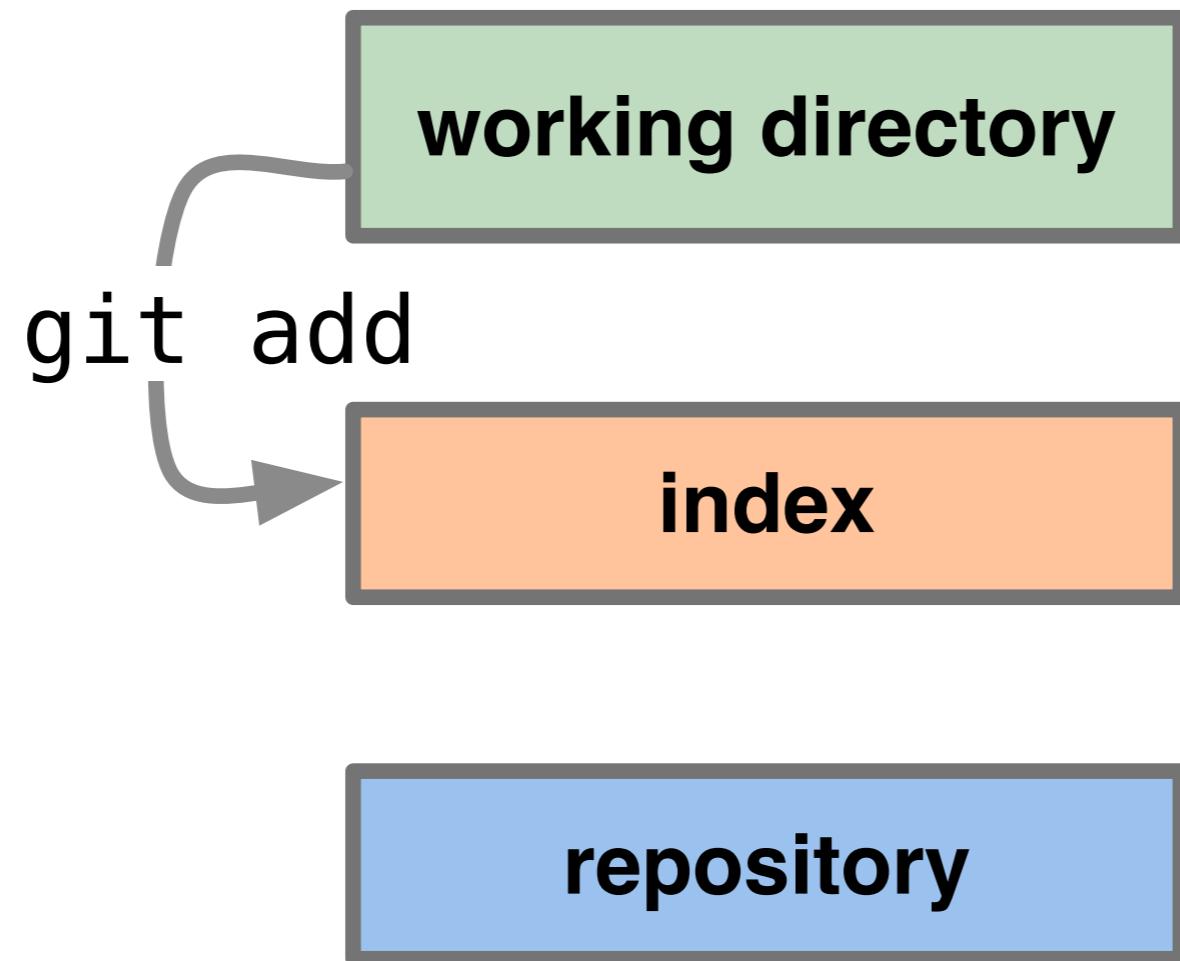
index

working directory

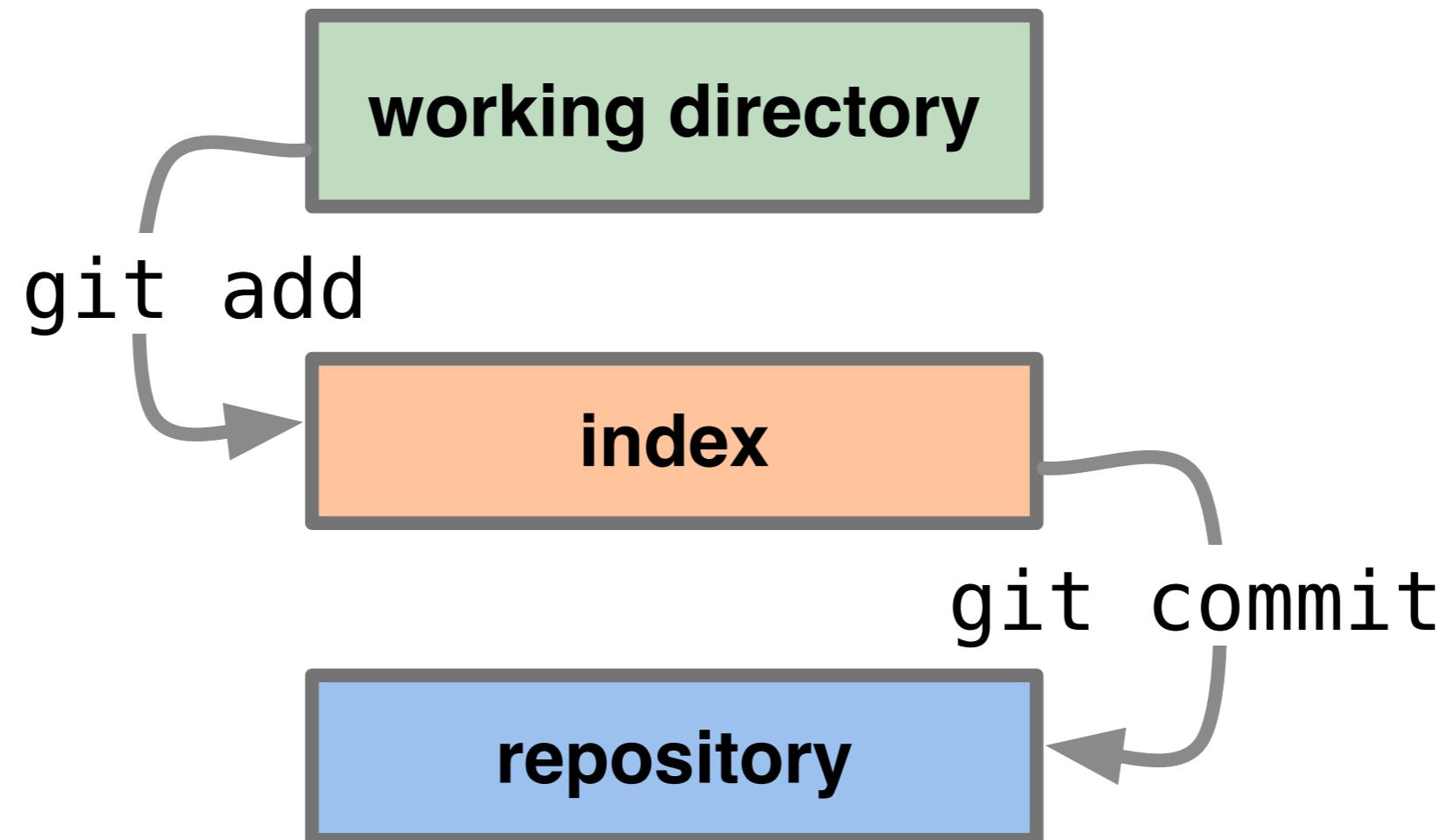
index

repository

index



index



what?

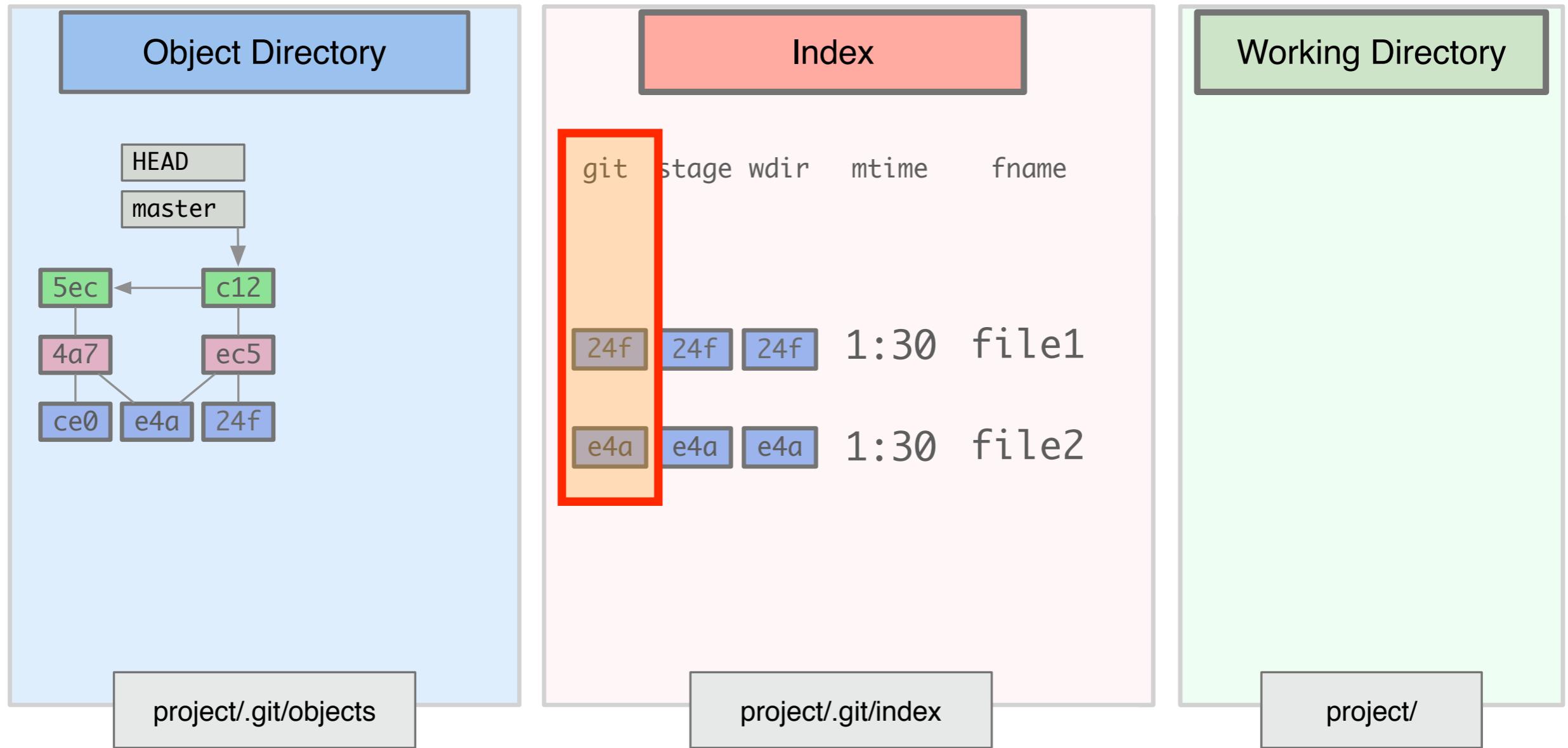




git checkout



git checkout



git checkout



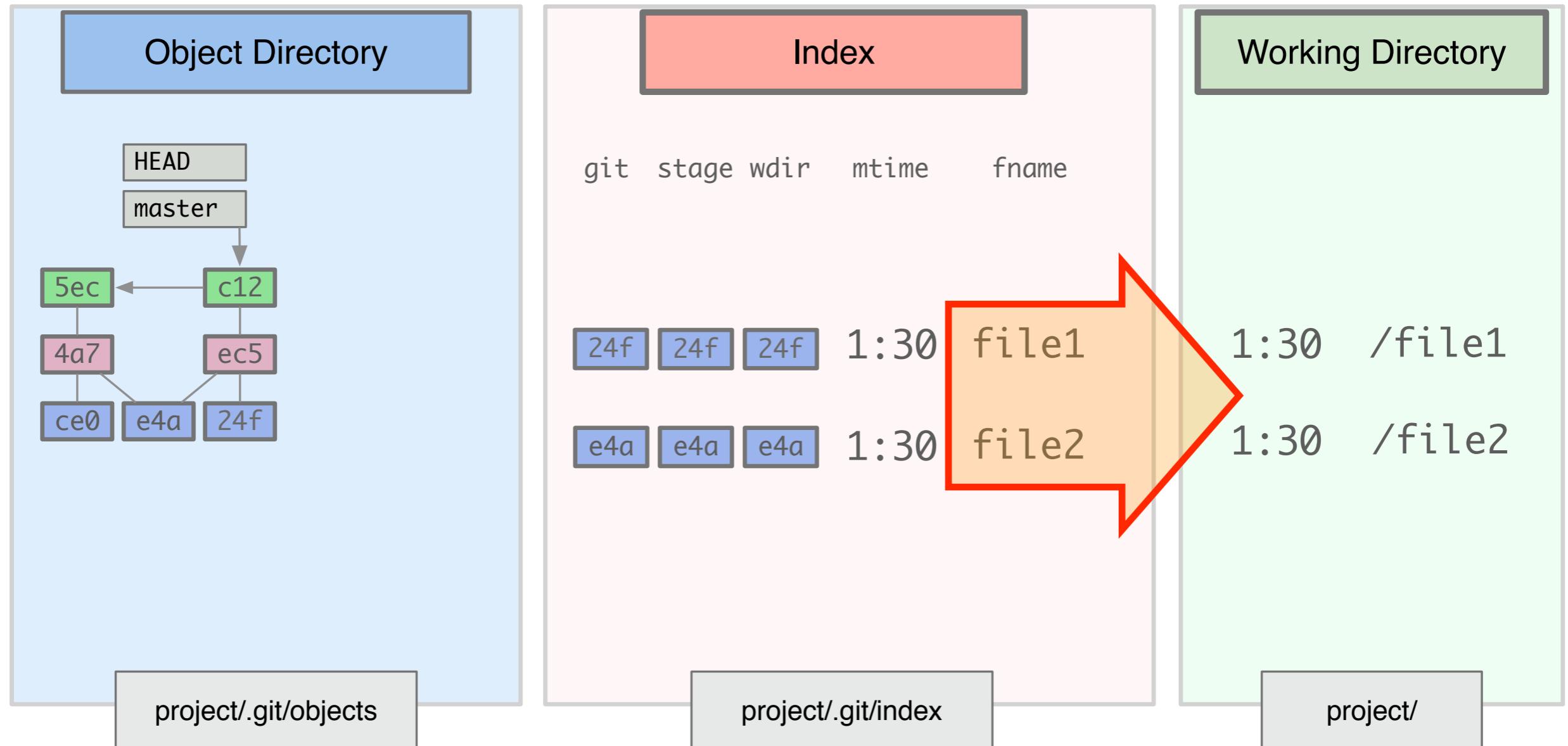
git checkout



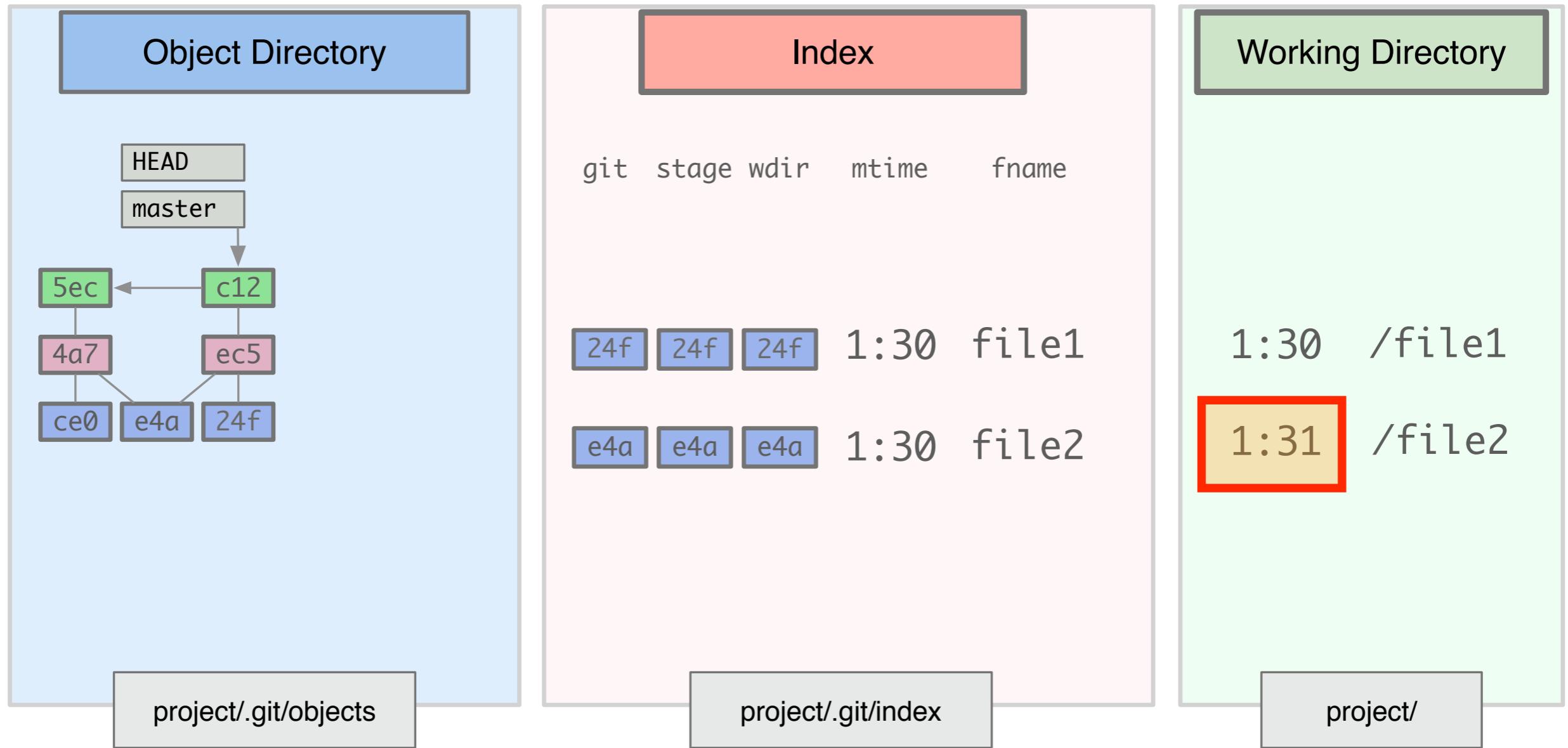
git checkout



git checkout



git checkout



vim file2

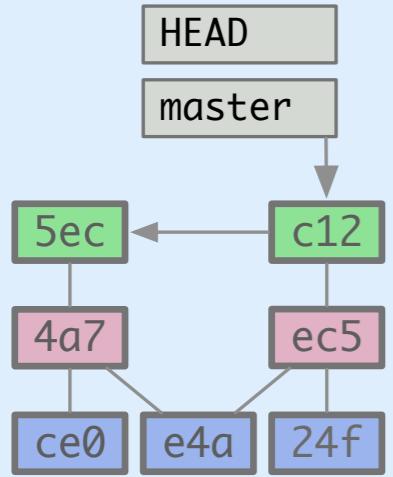


git status



git status

Object Directory



Index

git	stage	wdir	mtime	fname
24f	24f	24f	1:30	file1
e4a	e4a	1ba	1:31	file2

Working Directory

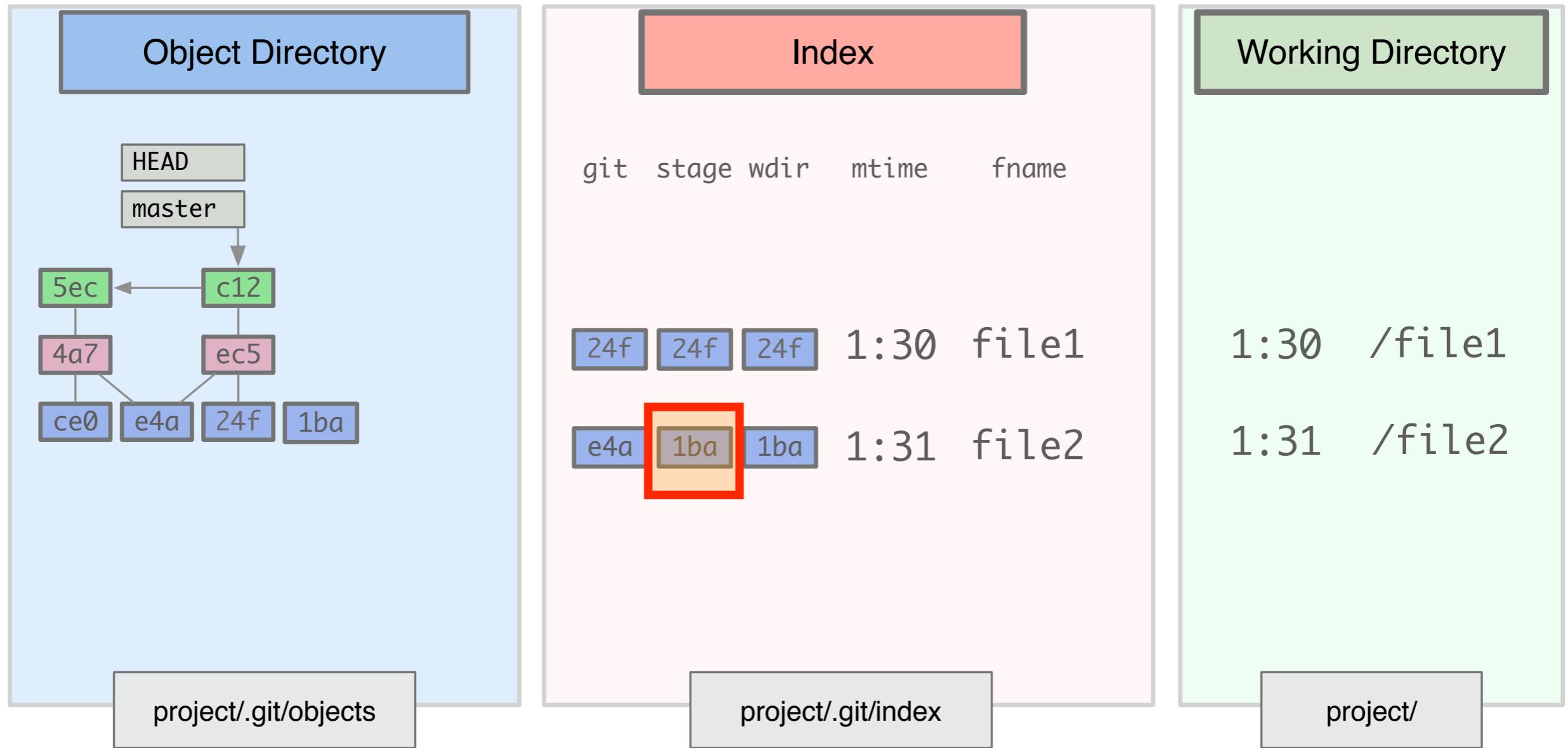
1:30 /file1
1:31 /file2

```
$>git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#           modified:   file2
#
```





git add file2



git add file2



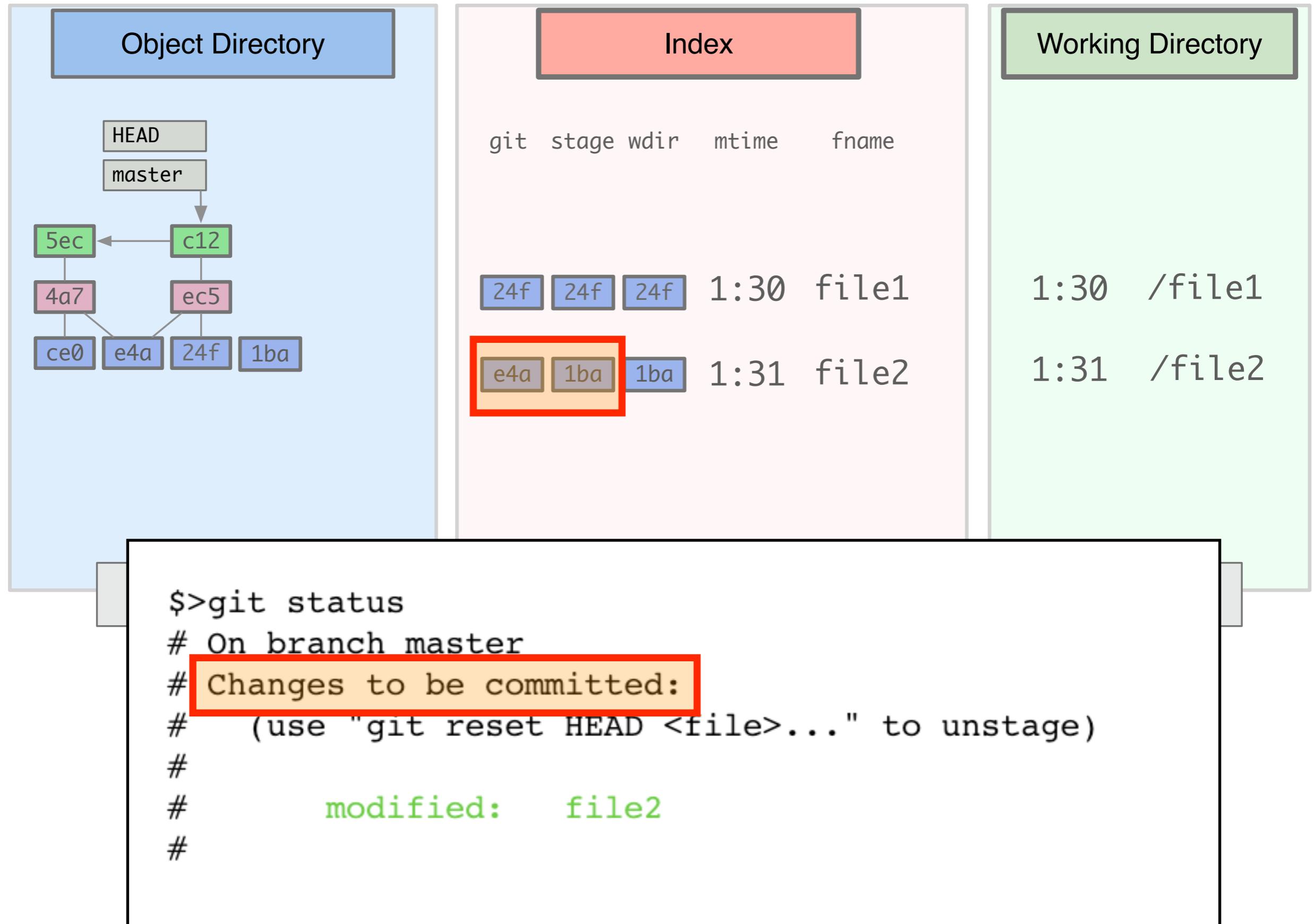
git add file2



git add file2

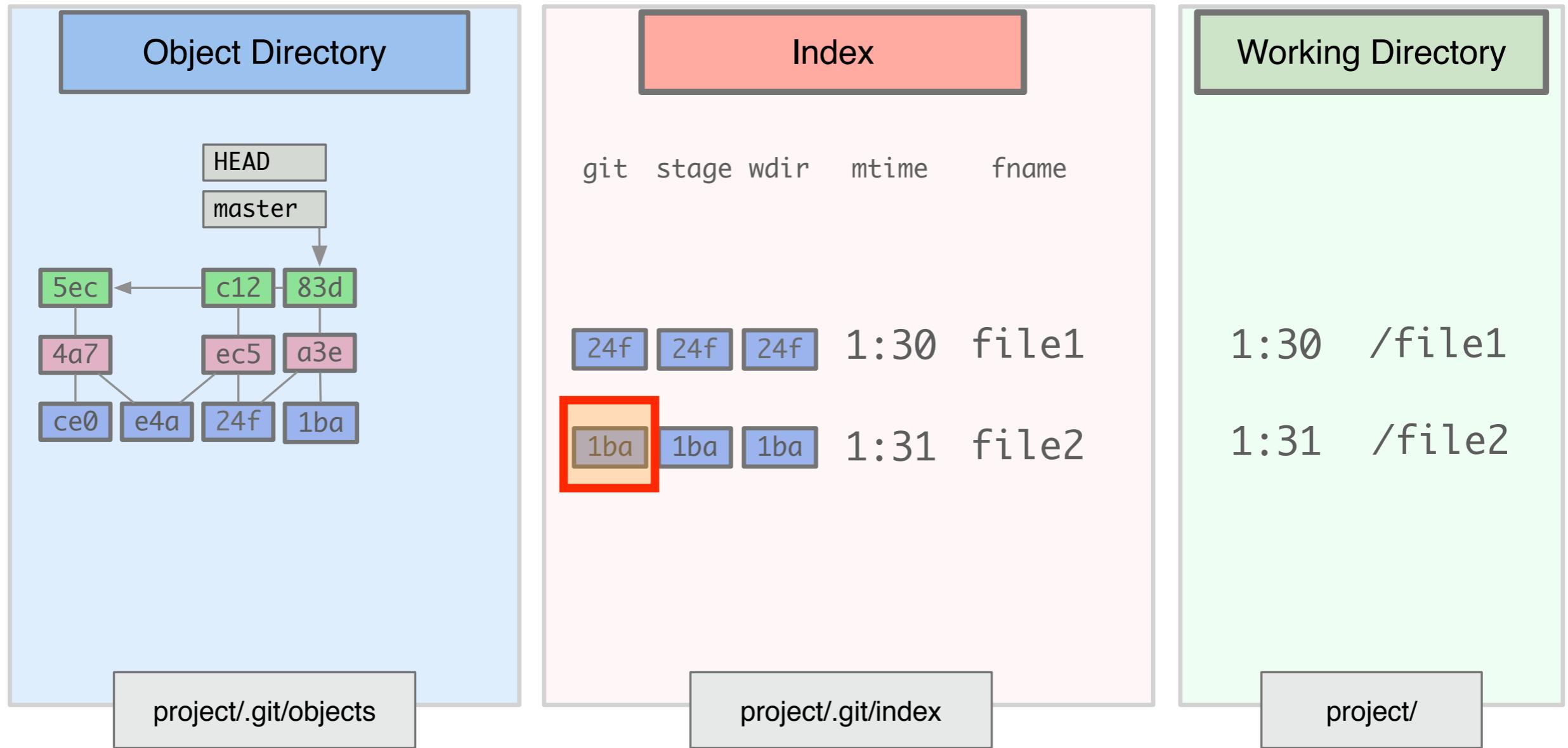


git status





git commit



git commit



git commit



git commit



vim file1 file2
git status



vim file1 file2
git status



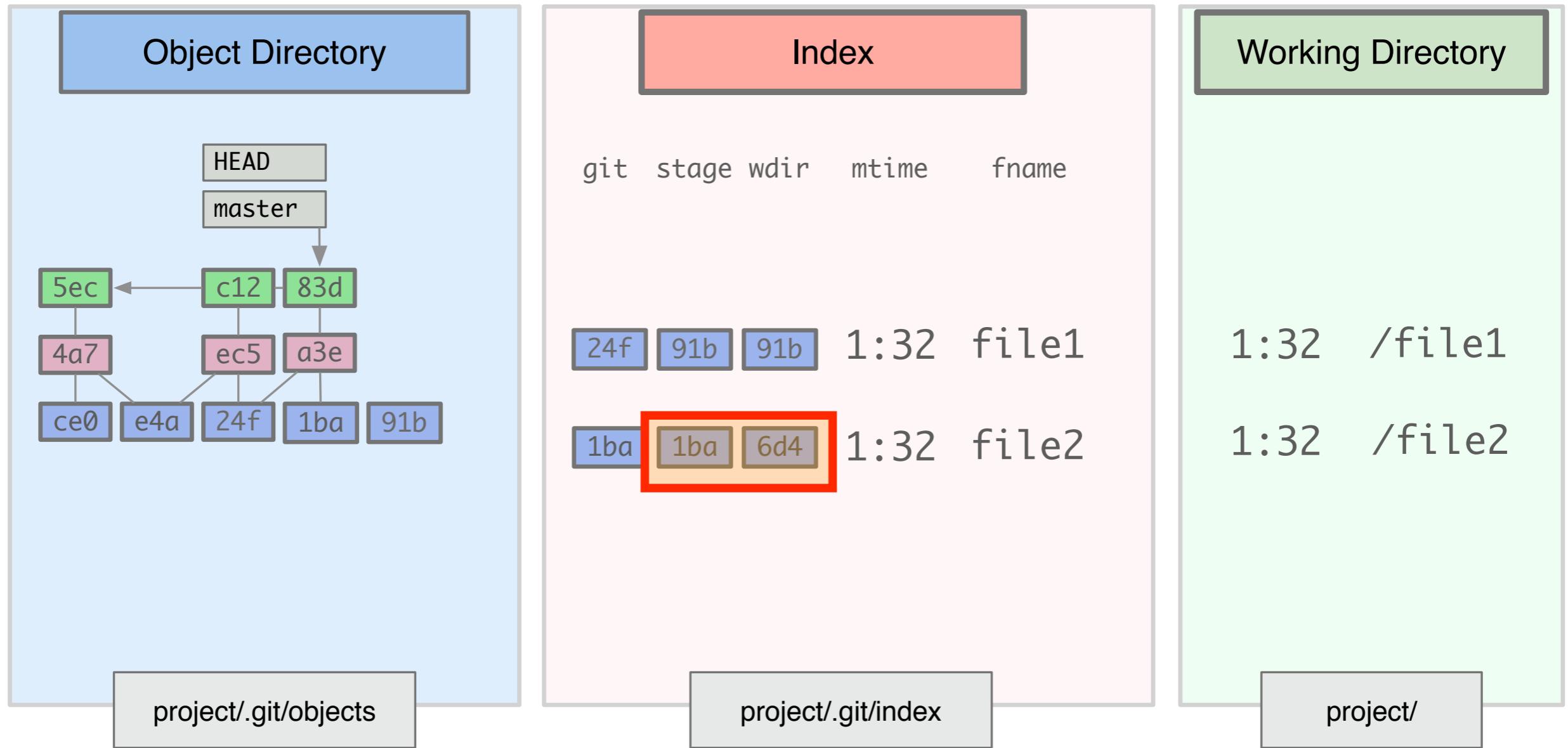
vim file1 file2
 git status



git add file1

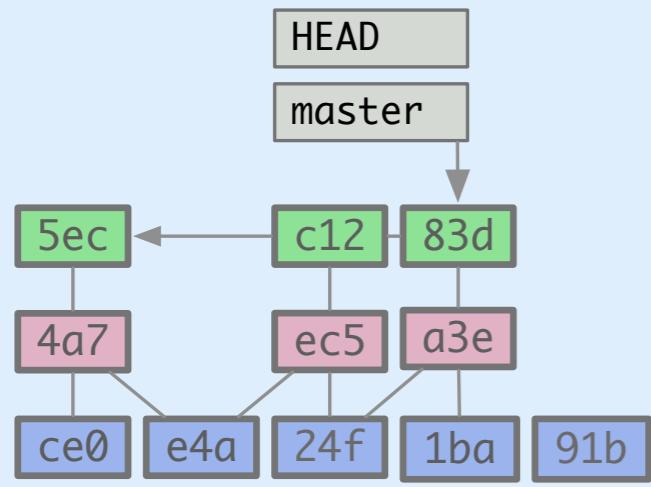


git add file1



git add file1

Object Directory



Index

git stage wdir mtime fname

24f	91b	91b	1:32	file1
1ba	1ba	6d4	1:32	file2

Working Directory

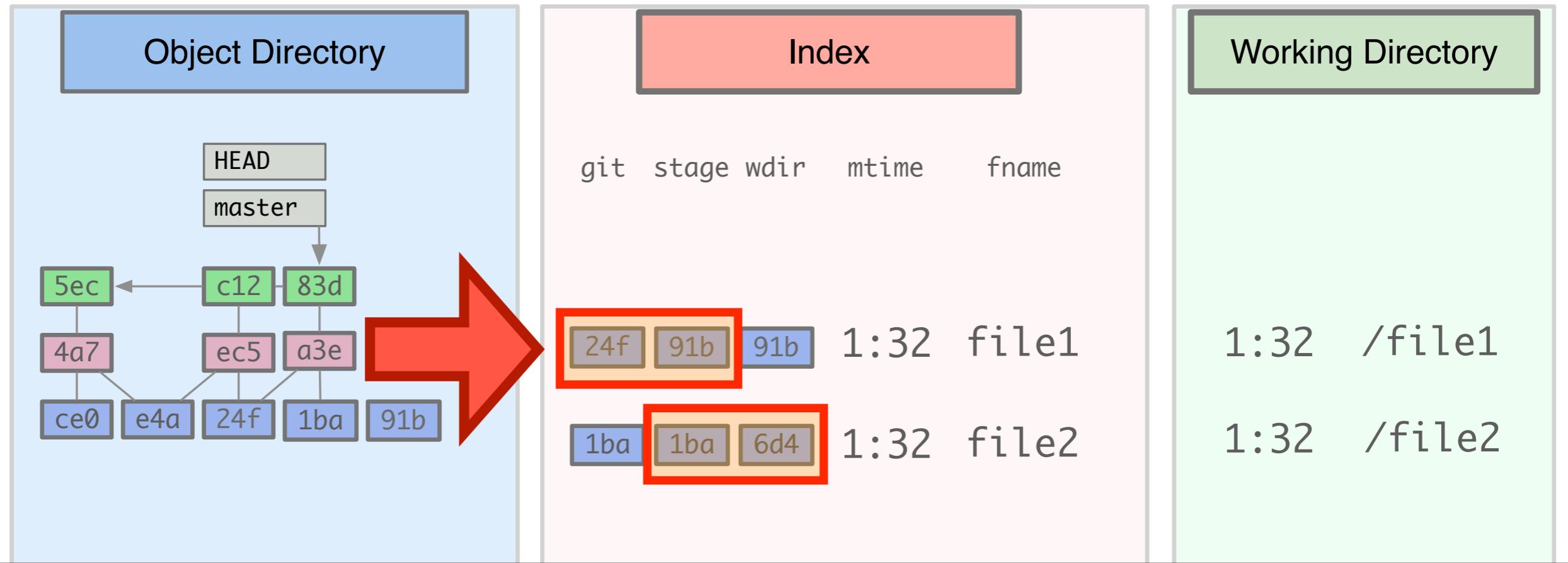
1:32 /file1
1:32 /file2

project/.git/objects

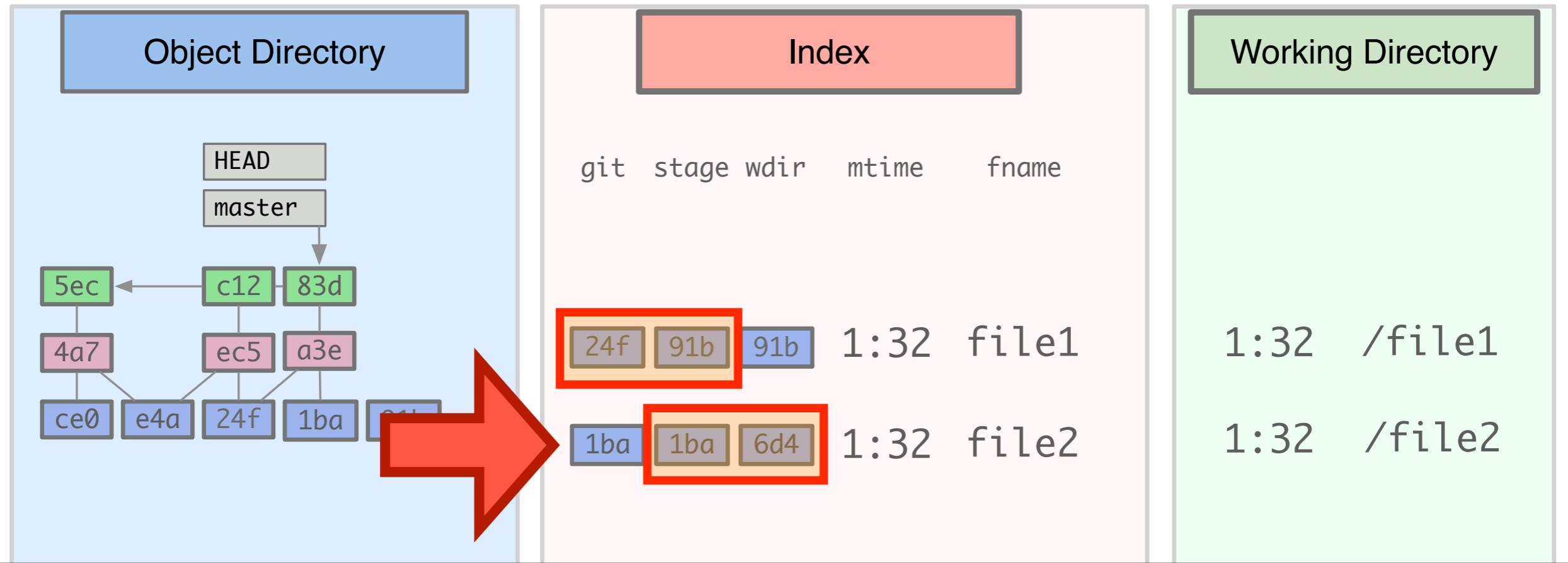
project/.git/index

project/

git status

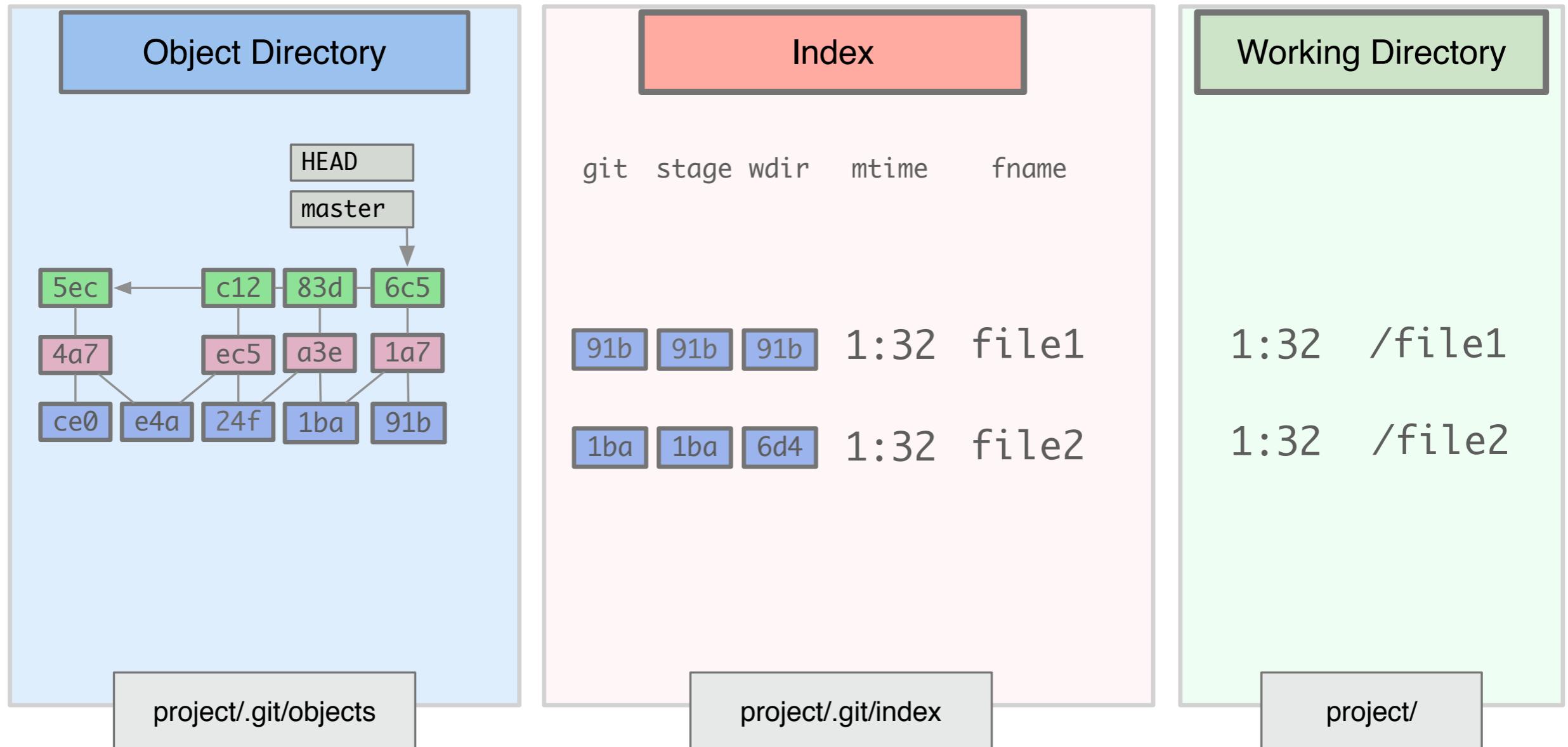


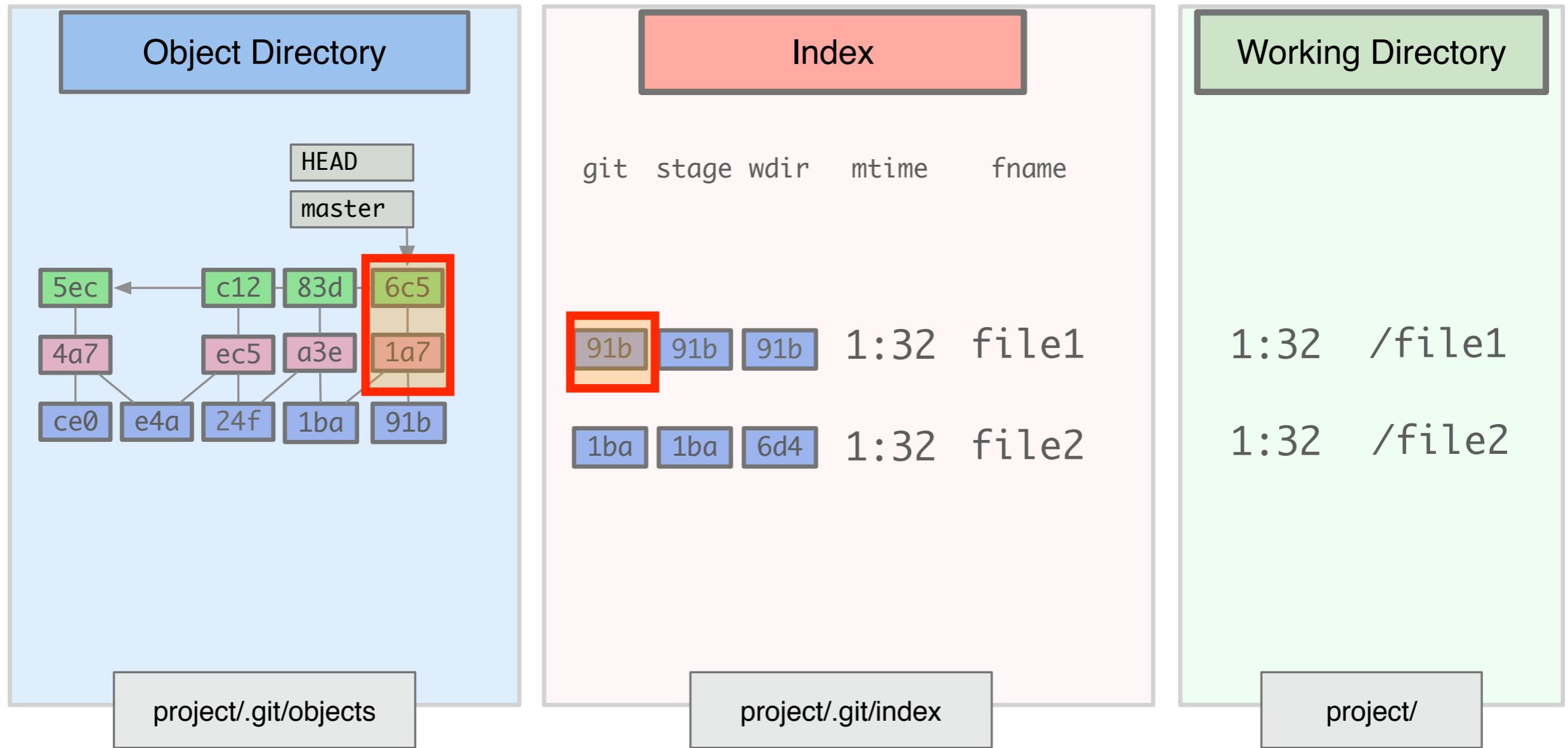
```
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#           modified:   file1
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#           modified:   file2
#
```



```
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   file1
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:   file2
#
```

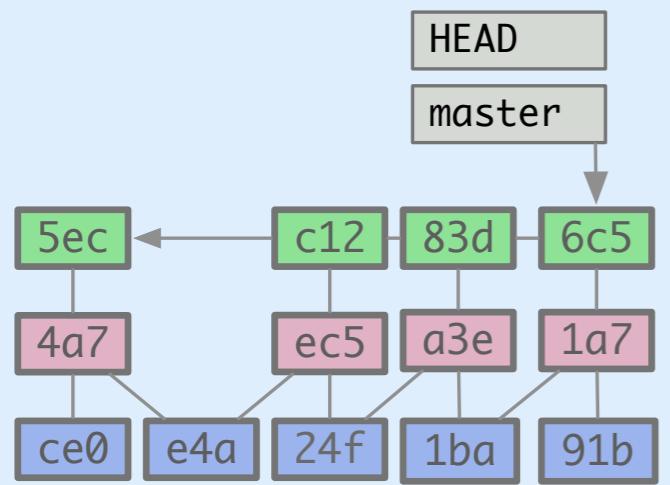
git commit





git commit

Object Directory



project/.git/objects

Index

git	stage	wdir	mtime	fname
91b	91b	91b	1:32	file1
1ba	1ba	6d4	1:32	file2

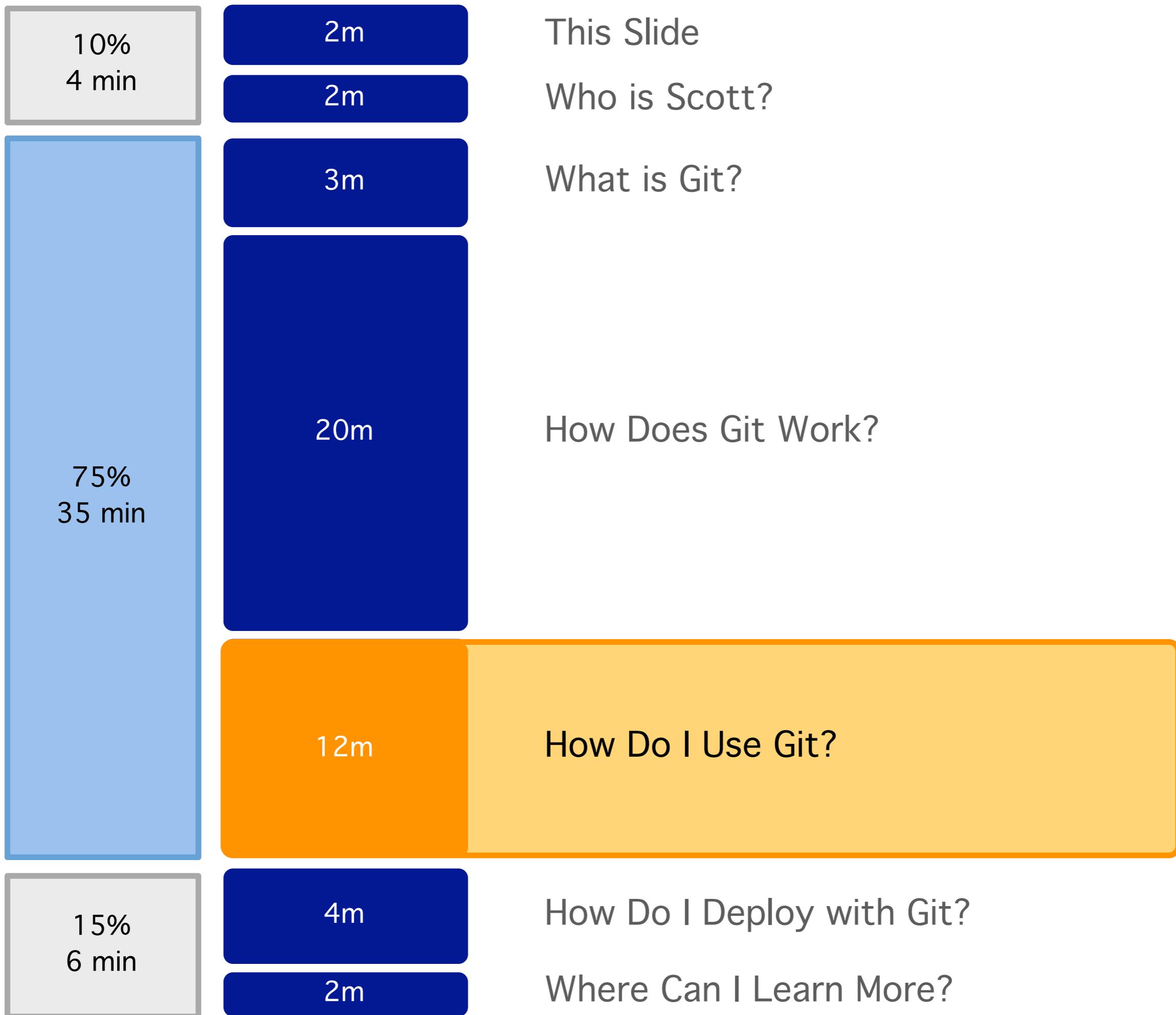
project/.git/index

Working Directory

1:32 /file1
1:32 /file2

project/

How Do I Use Git?



git-add	git-fast-export	git-merge-recur	git-revert
git-add--interactive	git-fast-import	git-merge-recursive	git-rm
git-am	git-fetch	git-merge-recursive-old	git-runstatus
git-annotate	git-fetch--tool	git-merge-resolve	git-send-email
git-apply	git-fetch-pack	git-merge-stupid	git-send-pack
git-applymbox	git-filter-branch	git-merge-subtree	git-sh-setup
git-applypatch	git-fmt-merge-msg	git-merge-tree	git-shell
git-archimport	git-for-each-ref	git-mergetool	git-shortlog
git-archive	git-format-patch	git-mktag	git-show
git-bisect	git-fsck	git-mktree	git-show-branch
git-blame	git-fsck-objects	git-mv	git-show-index
git-branch	git-gc	git-name-rev	git-show-ref
git-bundle	git-get-tar-commit-id	git-pack-objects	git-ssh-fetch
git-cat-file	git-grep	git-pack-redundant	git-ssh-pull
git-check-attr	git-gui	git-pack-refs	git-ssh-push
git-check-ref-format	git-hash-object	git-parse-remote	git-ssh-upload
git-checkout	git-http-fetch	git-patch-id	git-stash
git-checkout-index	git-http-push	git-peek-remote	git-status
git-cherry	git-imap-send	git-prune	git-stripSPACE
git-cherry-pick	git-index-pack	git-prune-packed	git-submodule
git-citool	git-init	git-pull	git-svn
git-clean	git-init-db	git-push	git-svnimport
git-clone	git-instaweb	git-quiltimport	git-symbolic-ref
git-commit	git-local-fetch	git-read-tree	git-tag
git-commit-tree	git-log	git-rebase	git-tar-tree
git-config	git-lost-found	git-rebase--interactive	git-unpack-file
git-convert-objects	git-ls-files	git-receive-pack	git-unpack-objects
git-count-objects	git-ls-remote	git-reflog	git-update-index
git-cvsexportcommit	git-ls-tree	git-relink	git-update-ref
git-cvsimport	git-mailinfo	git-remote	git-update-server-info
git-cvsserver	git-mailsplit	git-repack	git-upload-archive
git-daemon	git-merge	git-repo-config	git-upload-pack
git-describe	git-merge-base	git-request-pull	git-var
git-diff	git-merge-file	git-rerere	git-verify-pack
git-diff-files	git-merge-index	git-reset	git-verify-tag
git-diff-index	git-merge-octopus	git-resolve	git-web--browse
git-diff-stages	git-merge-one-file	git-rev-list	git-whatchanged
git-diff-tree	git-merge-ours	git-rev-parse	git-write-tree

152 Total Commands

git-add		git-revert
git-add--interactive		git-rm
git-am	git-fetch	
git-annotate		git-send-email
git-apply		
git-archive	git-format-patch	git-shortlog
git-bisect		git-show
git-blame		git-show-branch
git-branch	git-gc	git-mv
	git-grep	
	git-gui	
git-checkout		git-stash
		git-status
git-cherry-pick		git-submodule
git-citool	git-init	git-pull
git-clean		git-push
git-clone	git-instaweb	
git-commit		git-tag
git-config	git-log	git-rebase
		git-rebase--interactive
		git-remote
git-daemon	git-merge	
git-diff		

43 Porcelain Commands

git-add
git-add--interactive
git-am
git-annotate
git-apply

git-fetch

git-revert
git-rm

git-send-email

git-archive
git-bisect
git-blame
git-branch

git-format-patch

git-gc

git-grep
git-gui

git-mv

git-shortlog
git-show
git-show-branch

git-checkout

git-stash
git-status

git-cherry-pick
git-citool
git-clean
git-clone
git-commit

git-config

git-init

git-instaweb

git-log

git-pull
git-push

git-rebase
git-rebase--interactive

git-submodule

git-tag

git-daemon

git-diff

git-merge

git-remote

We'll Cover Half

getting git

git.or.cz

git.or.cz

wget
make
make install

yum install git-core

apt-get install git-core

port install git-core +svn

setup

```
git config --global user.name  
"Scott Chacon"
```

```
git config --global user.email  
"schacon@gmail.com"
```

```
$>cat ~/.gitconfig
[user]
    name = Scott Chacon
    email = schacon@gmail.com
[color]
    status = auto
    branch = auto
    interactive = auto
    diff = auto
```

```
$>cat ~/.gitconfig
[user]
    name = Scott Chacon
    email = schacon@gmail.com
[color]
    status = auto
    branch = auto
    interactive = auto
    diff = auto
```

git-add
git-add--interactive
git-am
git-annotate
git-apply

git-fetch

git-revert
git-rm

git-send-email

git-archive
git-bisect
git-blame
git-branch

git-format-patch

git-gc

git-grep
git-gui

git-mv

git-shortlog
git-show
git-show-branch

git-checkout

git-cherry-pick
git-citool
git-clean
git-clone
git-commit

git config

git-init

git-instaweb

git-log

git-pull
git-push

git-rebase
git-rebase--interactive

git-stash
git-status

git-submodule

git-tag

git-daemon

git-diff

git-merge

git-remote

| Down

getting a repo

git clone

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

Git Protocols

ssh://

http[s]://

git://

file:///

rsync://

Git Protocols

ssh://

http[s]://

git://

file:///

~~rsync://~~

Git Protocols

ssh://

push

http[s]://

git://

file:///

~~rsync://~~

Git Protocols

pull

ssh://

http[s]://

git://

file:///

~~rsync://~~

git clone

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

git clone

```
$>git clone git://github.com/schacon/ticgit.git
Initialized empty Git repository in /Users/schacon/examples
/ticgit/.git/
remote: Generating pack...
remote: Done counting 483 objects.
remote: Deltifying 483 objects...
remote: 100% (483/483) done
remote: Total 483 (delta 198), reused 483 (delta 198)
Receiving objects: 100% (483/483), 60.15 KiB, done.
Resolving deltas: 100% (198/198), done.
$>_
```

git clone

```
$>git clone git://github.com/schacon/ticgit.git
Initialized empty Git repository in /Users/schacon/examples
/ticgit/.git/
remote: Generating pack...
remote: Done counting 483 objects.
remote: Deltifying 483 objects...
remote: 100% (483/483) done
remote: Total 483 (delta 198), reused 483 (delta 198)
Receiving objects: 100% (483/483), 60.15 KiB, done.
Resolving deltas: 100% (198/198), done.
$>_
```

git clone

```
$>git clone git://github.com/schacon/ticgit.git
Initialized empty Git repository in /Users/schacon/examples
/ticgit/.git/
remote: Generating pack...
remote: Done counting 483 objects.
remote: Deltifying 483 objects...
remote: 100% (483/483) done
remote: Total 483 (delta 198), reused 483 (delta 198)
Receiving objects: 100% (483/483), 60.15 KiB, done.
Resolving deltas: 100% (198/198), done.
$>_
```

initializing a repo

```
cd myproject
```

```
git init
```

```
git add .
```

```
git commit
```

```
$>git init
```

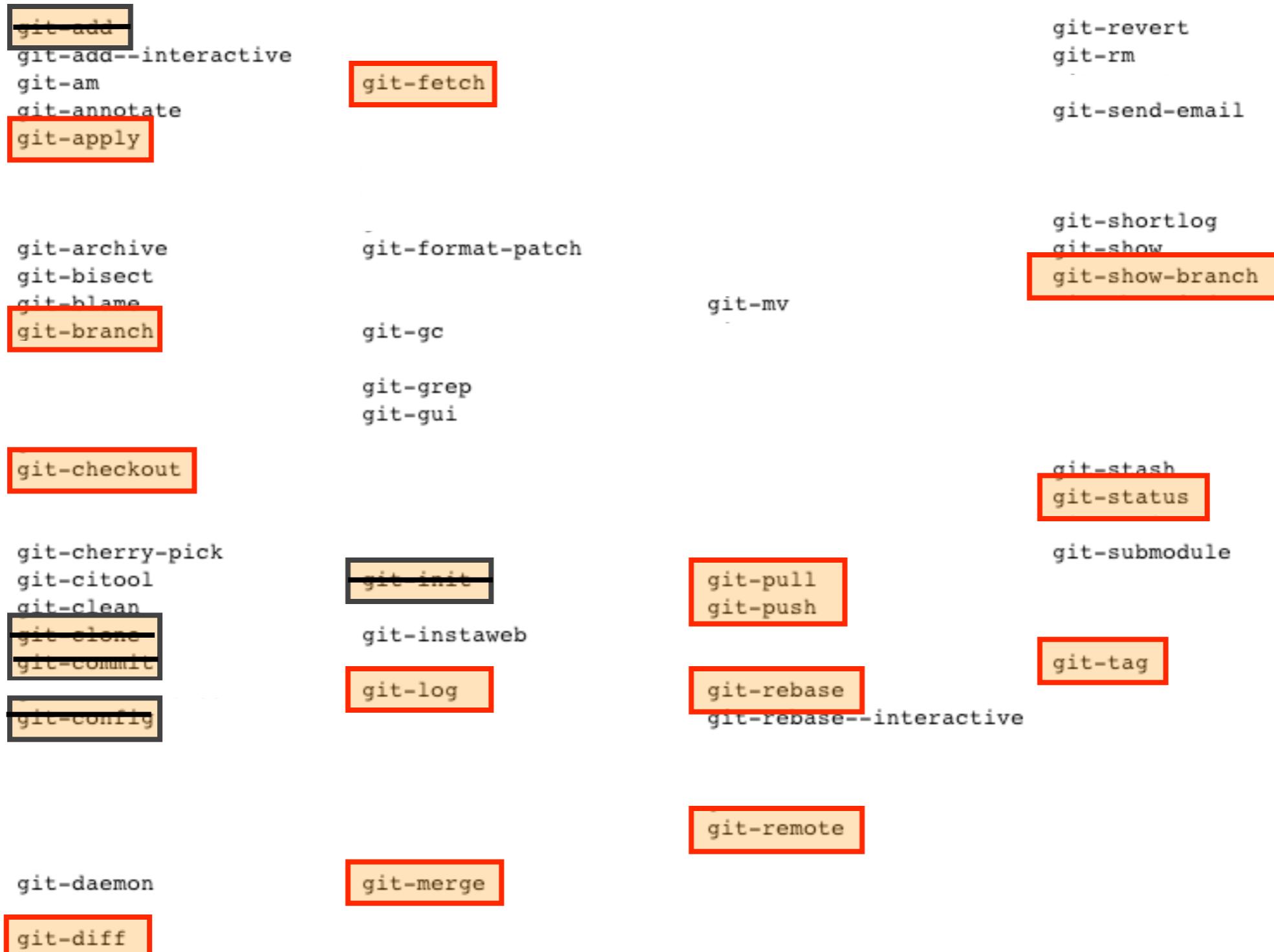
```
$>git init  
Initialized empty Git repository in .git/
```

```
$>git init  
Initialized empty Git repository in .git/  
$>git add .
```

```
$>git init  
Initialized empty Git repository in .git/  
$>git add .  
$>git commit -m 'my first commit'
```

```
$>git init
Initialized empty Git repository in .git/
$>git add .
$>git commit -m 'my first commit'
Created initial commit bfe09f9: my first commit
 3 files changed, 3 insertions(+), 0 deletions(-)
create mode 100644 README
create mode 100644 Rakefile
create mode 100644 lib/mylib.rb
$>_
```

```
$>git init
Initialized empty Git repository in .git/
$>git add .
$>git commit -m 'my first commit'
Created initial commit bfe09f9: my first commit
 3 files changed, 3 insertions(+), 0 deletions(-)
create mode 100644 README
create mode 100644 Rakefile
create mode 100644 lib/mylib.rb
$>_
```



5 Down

normal workflow

.gitignore

```
$>cat .gitignore
tmp/**/*
log/*
config/database.yml
config/environments/production.rb
.DS_Store
```

```
$>cat .gitignore
tmp/**/*
log/*
config/database.yml
config/environments/production.rb
.DS_Store
```

adding and committing

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

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

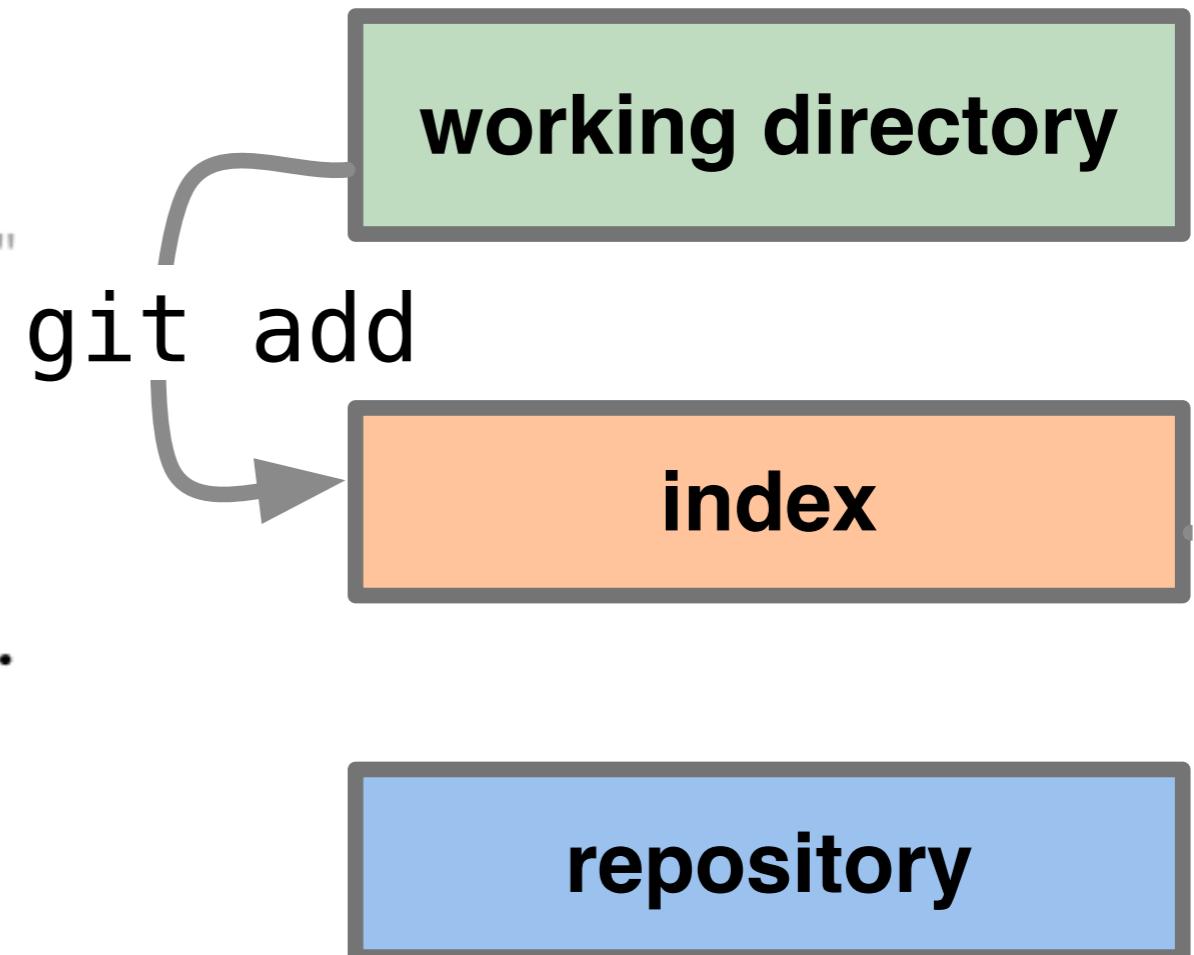
```
$>git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:   README
#       modified:   TODO
#
no changes added to commit (use "git add" and/or "git commit -a")
$>git add README
```

```
$>git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:    README
#       modified:    TODO
#
no changes added to commit (use "git add" and/or "git commit -a")
$>git add README
$>git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:    README
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:    TODO
#
$>_
```

```
$>git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:    README
#       modified:    TODO
#
no changes added to commit (use "git add" and/or "git commit -a")
$>git add README
$>git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:    README
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:    TODO
#
$>_
```

```
$>git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:    README
#       modified:    TODO
#
no changes added to commit (use "git add README" to update what will be committed)

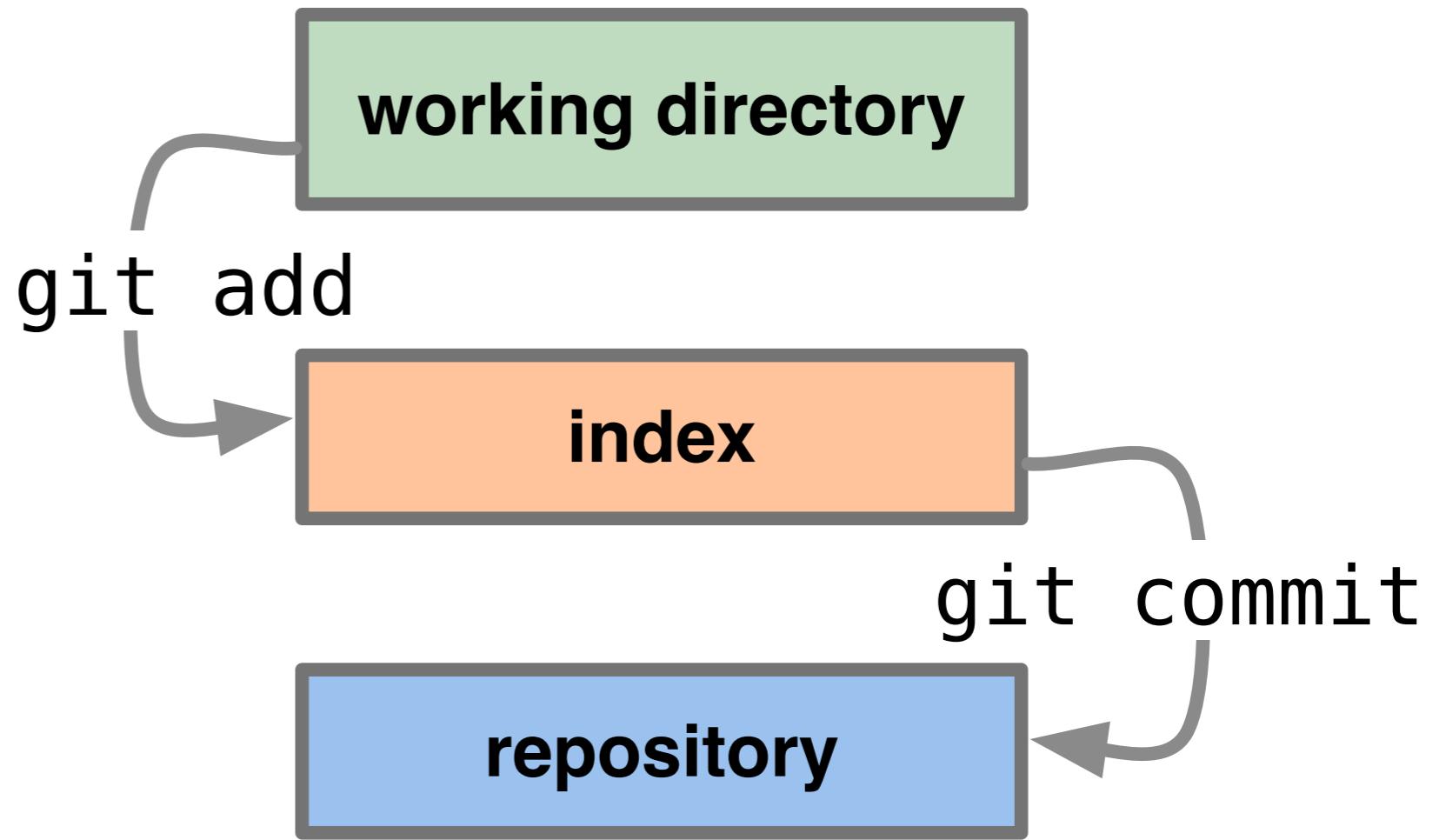
$>git add README
$>git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>.."
#
#       modified:    README
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:    TODO
#
$>_
```



```
$>git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   README
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:   TODO
#
$>git commit -m 'updated README'
Created commit c4f8fd6: updated README
 1 files changed, 1 insertions(+), 0 deletions(-)
$>_
```

```
$>git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       modified:   README
#
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:   TODO
#
$>git commit -m 'updated README'
Created commit c4f8fd6: updated README
  1 files changed, 1 insertions(+), 0 deletions(-)
$>git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:   TODO
#
no changes added to commit (use "git add" and/or "git commit -a")
$>
```

```
$>git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD
#
#       modified:   README
#
# Changed but not updated
#   (use "git add <file>.
#
#       modified:   TODO
#
$>git commit -m 'updated
Created commit c4f8fd6: u
 1 files changed, 1 inser
$>git status
# On branch master
# Changed but not updated:
#   (use "git add <file>..." to update what will be committed)
#
#       modified:   TODO
#
no changes added to commit (use "git add" and/or "git commit -a")
$>
```



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
```

git log --pretty

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

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

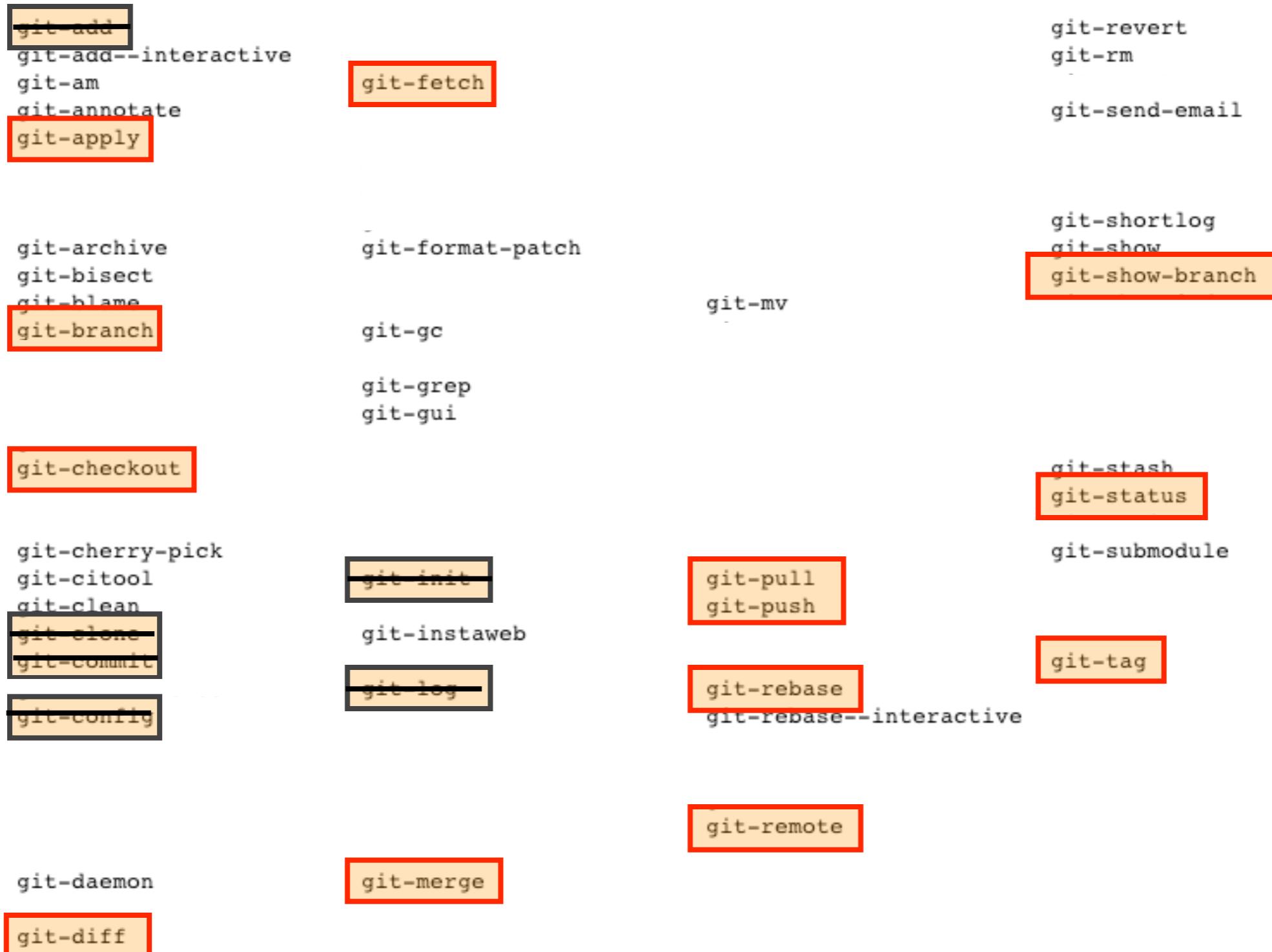
```
3e625e2139a71bfbaa85b055c139bceafab7e2c0 updated rakefile
4fefbaafd51ff3e79b84e9ee1a0dac2848e21a98 commit the todo
c4f8fd68ae3897d1d10456ed2b93d87cdc53d6e8 updated README
a5f4a0daa9b13bb85283461ddaba6e589e34ccb5 added cat-file
310154e3c7db47d8bac935c2c43aee6afac11aae updated README formatting
f7f3f6dd8fd3fa40f052427c32785a0fa01aaa5f 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 functio
47c668a9e8f93f65ff9019c587f3b2204bd6c98d fixed conflict
ca28a28e64192af4527ea63665dceaa9bdb4a4b 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
```



6 Down

git diff

git diff (treeish1) (treeish2)

```
diff --git a/.gitignore b/.gitignore
index 224f138..7e56ec7 100644
--- a/.gitignore
+++ b/.gitignore
@@ -1,4 +1,3 @@
 bin/*.log
 pkg
 coverage
-ticgit*gem
diff --git a/Rakefile b/Rakefile
index b30e89c..bd7b91a 100644
--- a/Rakefile
+++ b/Rakefile
@@ -31,7 +31,6 @@ desc "Clean out the coverage and pkg directories"
 task :clean do
   rm_rf 'coverage'
   rm_rf 'pkg'
-  rm Dir.glob('ticgit*gem')
end

task :default => "pkg/#{spec.name}-#{spec.version}.gem" do
diff --git a/ticgit.gemspec b/ticgit.gemspec
index c0e96b7..775eaf1 100644
--- a/ticgit.gemspec
+++ b/ticgit.gemspec
@@ -6,7 +6,7 @@ Gem::Specification.new do |s|
  s.author      = "Scott Chacon"
  s.email       = "schacon@gmail.com"
  s.summary     = "A distributed ticketing system for Git projects."
-  s.files      = ["lib/ticgit/base.rb", "lib/ticgit/cli.rb", "lib/ticgit/comment.rb"]
+  s.files      = ["lib/ticgit", "lib/ticgit/base.rb", "lib/ticgit/cli.rb", "lib/ticg
  s.bindir      = 'bin'
  s.executables << "ti"
```

```
git diff master^ > change.diff
```

```
git diff master^ > change.diff
```

```
patch -p1 < change.diff
```

```
git diff master^ > change.diff
```

```
patch -p1 < change.diff
```

or

```
git apply change.diff
```

git and patches

Single Commit

```
git format-patch origin/master --stdout >  
story.patch
```

```
git am < story.patch
```

Multiple Commits

```
$ git format-patch -o patches origin  
$ git send-email --to list@email.com patches  
  
$ git am mbox
```

Multiple Commits

1 patch file per commit
in this directory



```
$ git format-patch -o patches origin
$ git send-email --to list@email.com patches

$ git am mbox
```

example: patching rails

Patching Rails

```
git clone git://github.com/rails/rails.git
```

```
cd rails; (vim/emacs etc)
```

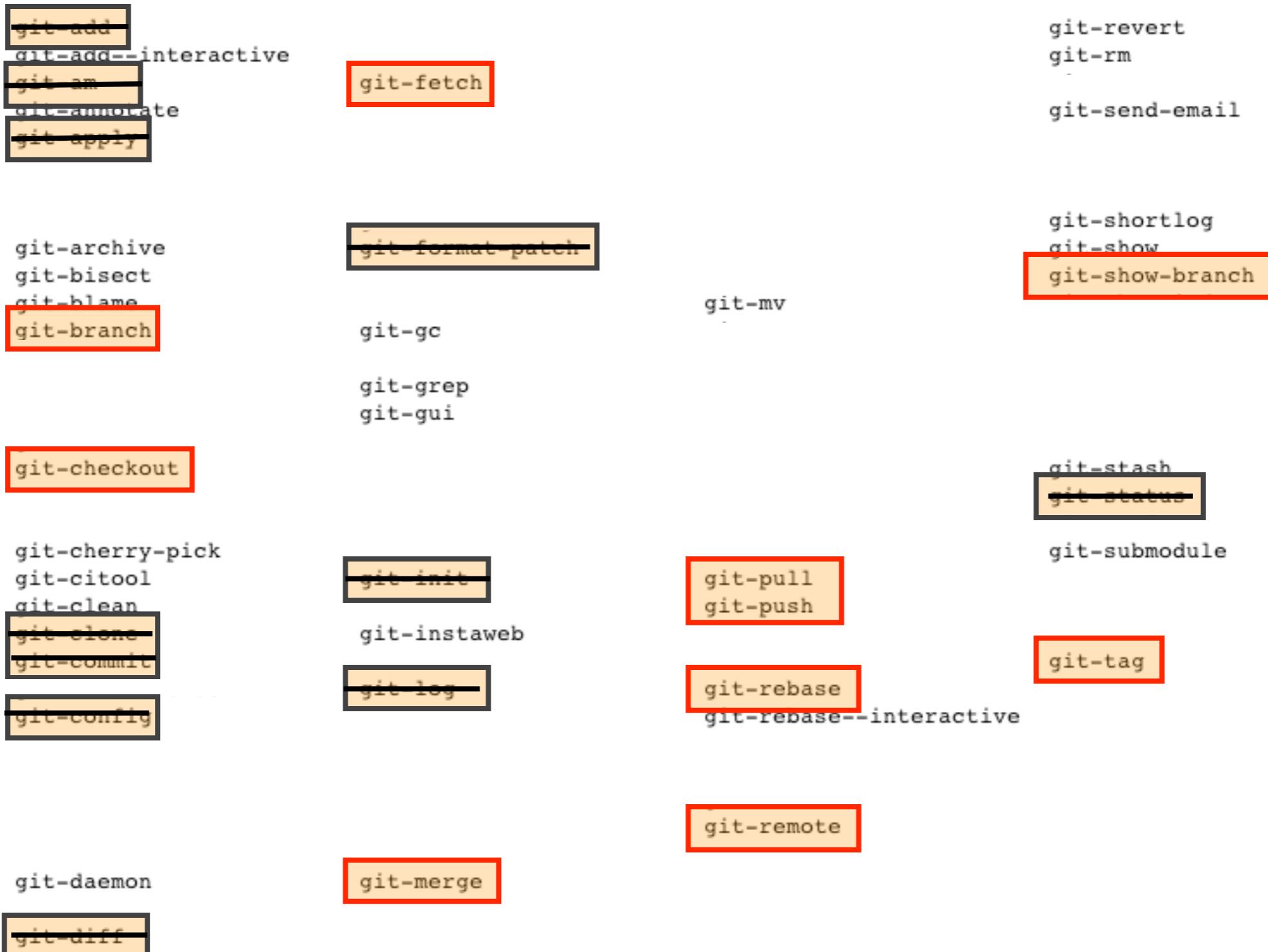
```
git format-patch origin/master --stdout > story.patch
```

```
upload story.patch to rails.lighthouseapp.com
```

Applying a Rails Patch

download story.patch from
rails.lighthouseapp.com

```
git am < story.patch
```



|| Down

branch, merge, rebase

master



master



master



develop

master

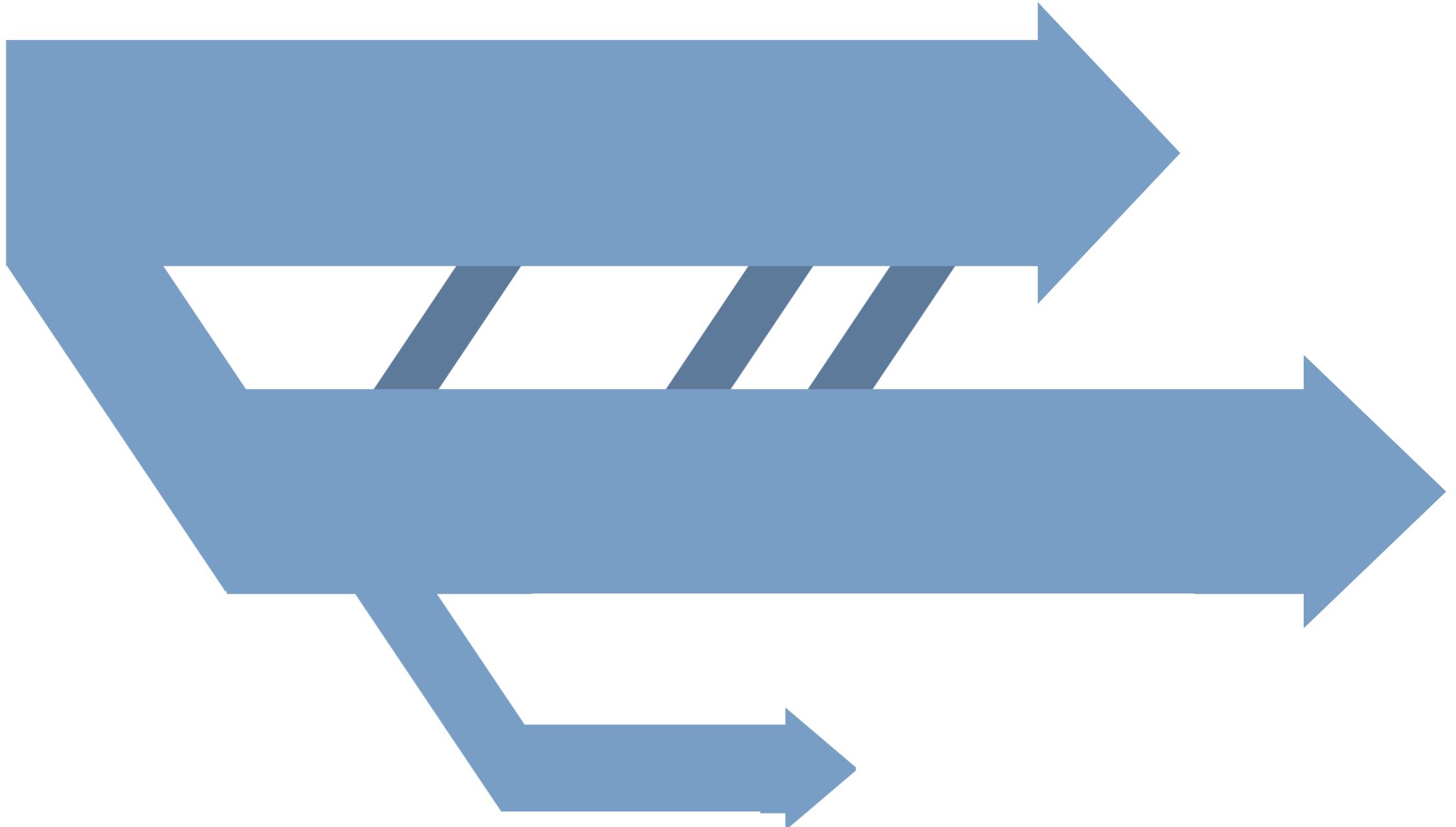
develop



master

develop

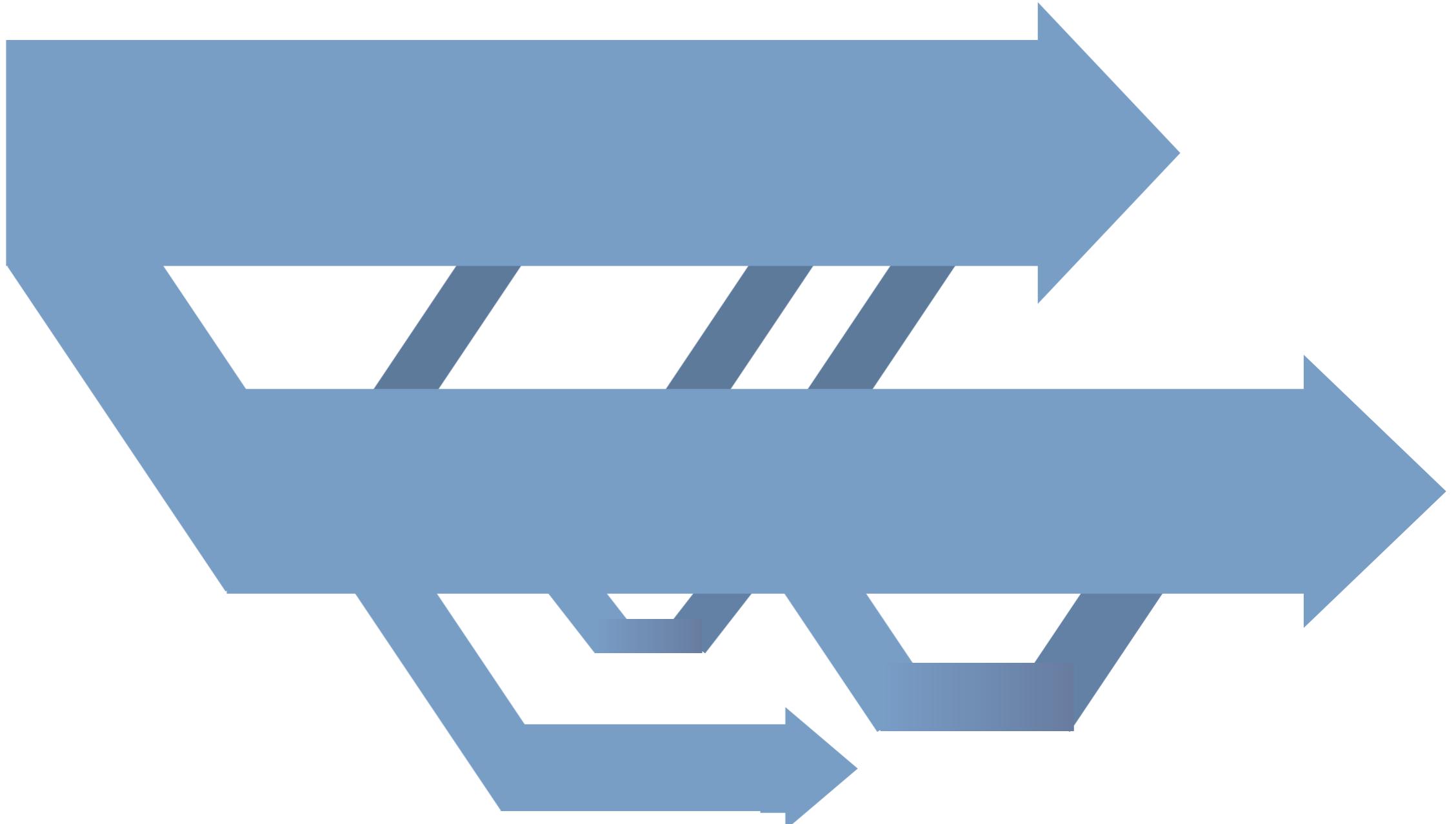
topic



master

develop

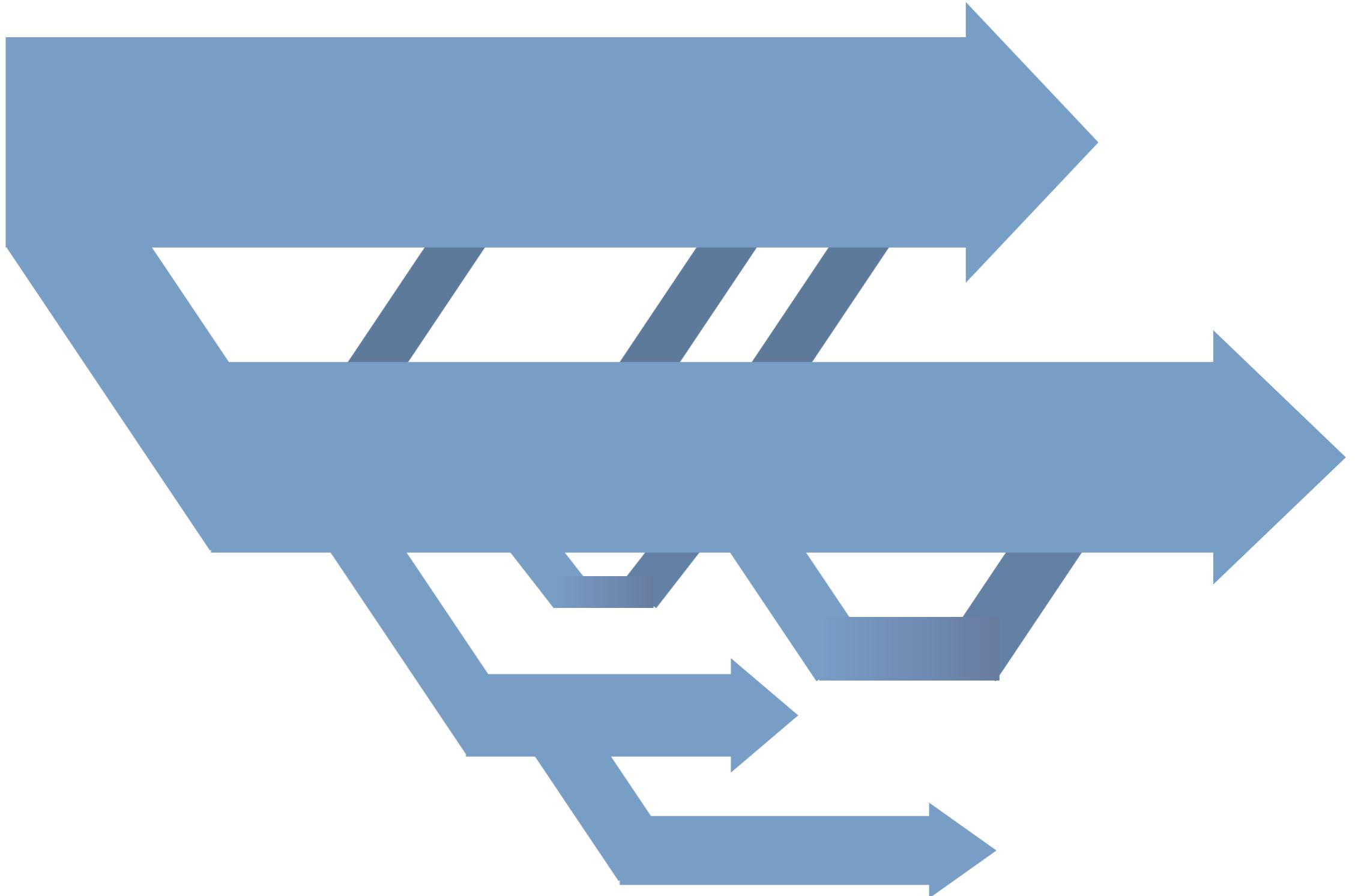
topic



master

develop

topic



master

“production”

develop

topic

master

“production”

develop

“trunk”

topic

master

“production”

develop

“trunk”

topic

convention!

branching workflow

normal development

want to try an idea

have to do a hotfix

merge idea into development

push to production

branching workflow

normal development

want to try an idea

have to do a hotfix

merge idea into development

push to production

```
$>git branch
```

```
$>git branch
```

```
* develop
```

```
 master
```

```
 story95
```

```
$>
```

```
_
```

```
$>git branch  
* develop  
  master  
  story95  
$>git show-branch
```

```
$>git branch
* develop
  master
  story95
$>git show-branch
* [develop] new file_size function
! [master] added cat-file
! [story95] show-commit-tree added
---
+ [story95] show-commit-tree added
+ [story95^] added revparse
* [develop] new file_size function
* [develop^] modified show to raw
*++ [master] added cat-file
$>_
```

```
$>git branch
* develop
  master
  story95
$>git show-branch
* [develop] new file_size function
! [master] added cat-file
! [story95] show-commit-tree added
---
+ [story95] show-commit-tree added
+ [story95^] added revparse
* [develop] new file_size function
* [develop^] modified show to raw
*++ [master] added cat-file
$>_
```

```
$>git branch
* develop
  master
  story95
$>git show-branch
* [develop] new file_size function
! [master] added cat-file
! [story95] show-commit-tree added
---
+ [story95] show-commit-tree added
+ [story95^] added revparse
* [develop] new file_size function
* [develop^] modified show to raw
*++ [master] added cat-file
$>_
```

```
$>git branch
* develop
  master
  story95
$>git show-branch
* [develop] new file_size function
! [master] added cat-file
! [story95] show-commit-tree added
---
+ [story95] show-commit-tree added
+ [story95^] added revparse
* [develop] new file_size function
* [develop^] modified show to raw
*++ [master] added cat-file
$>_
```

```
$>git branch
```

```
* develop
```

```
 master
```

```
 story95
```

```
$>git show-branch
```

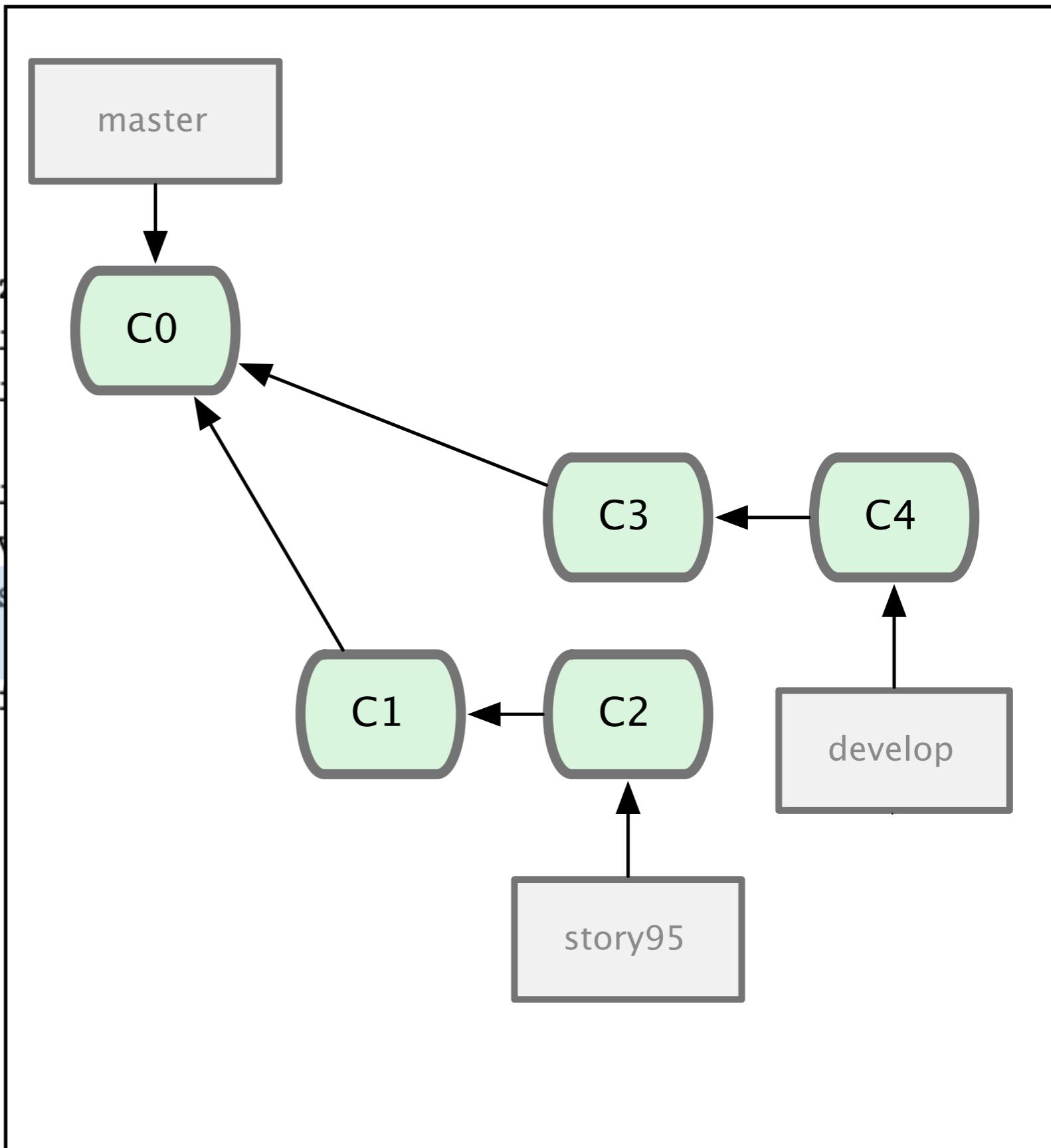
```
* [develop] new file_size  
! [master] added cat-f  
! [story95] show-comm
```

```
---  
+ [story95] show-comm  
+ [story95^] added rev
```

```
* [develop] new file_s  
* [develop^] modified
```

```
*++ [master] added cat-
```

```
$>
```



branching workflow

normal development

want to try an idea

have to do a hotfix

merge idea into development

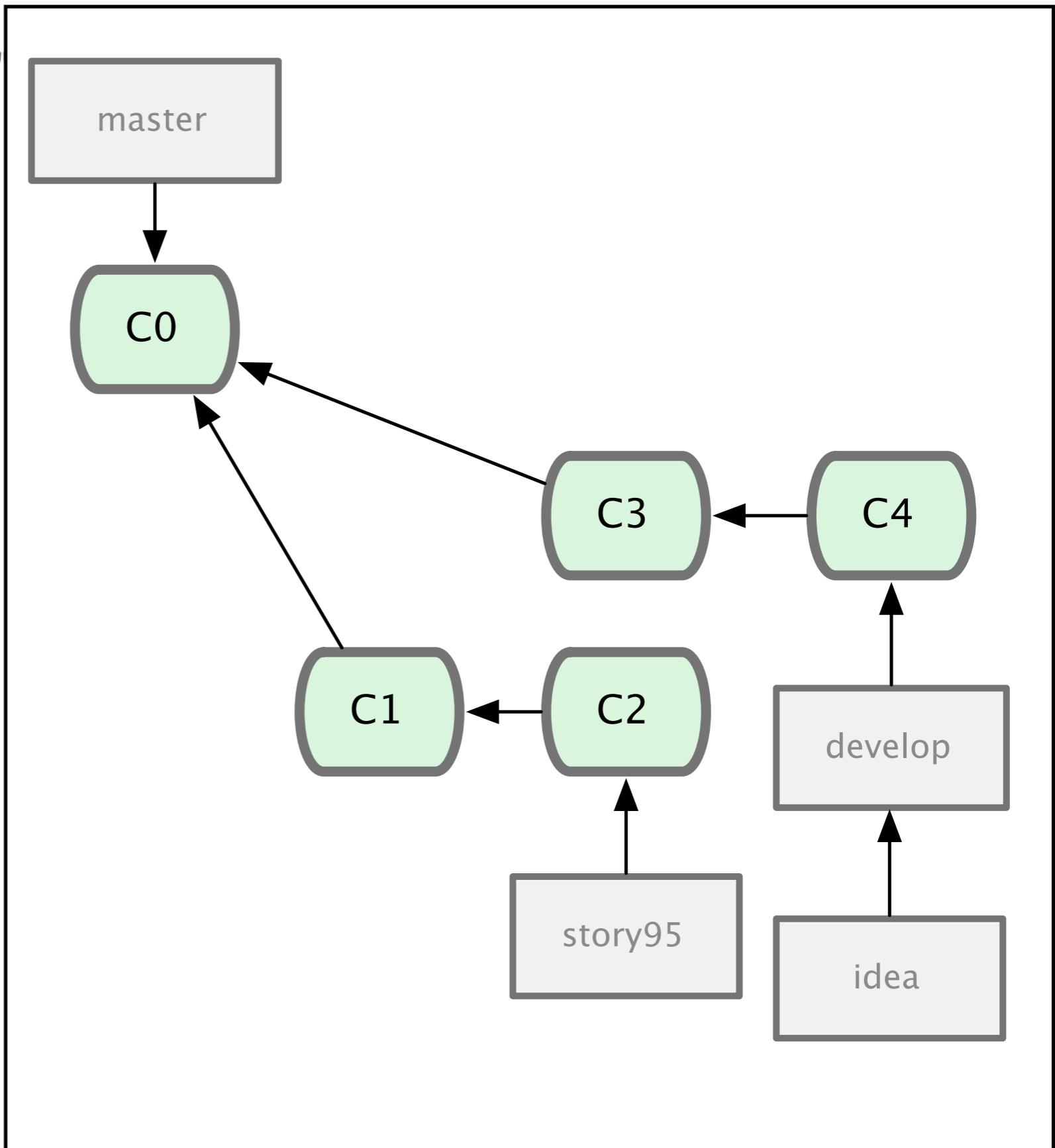
push to production

```
$>git checkout -b idea
```

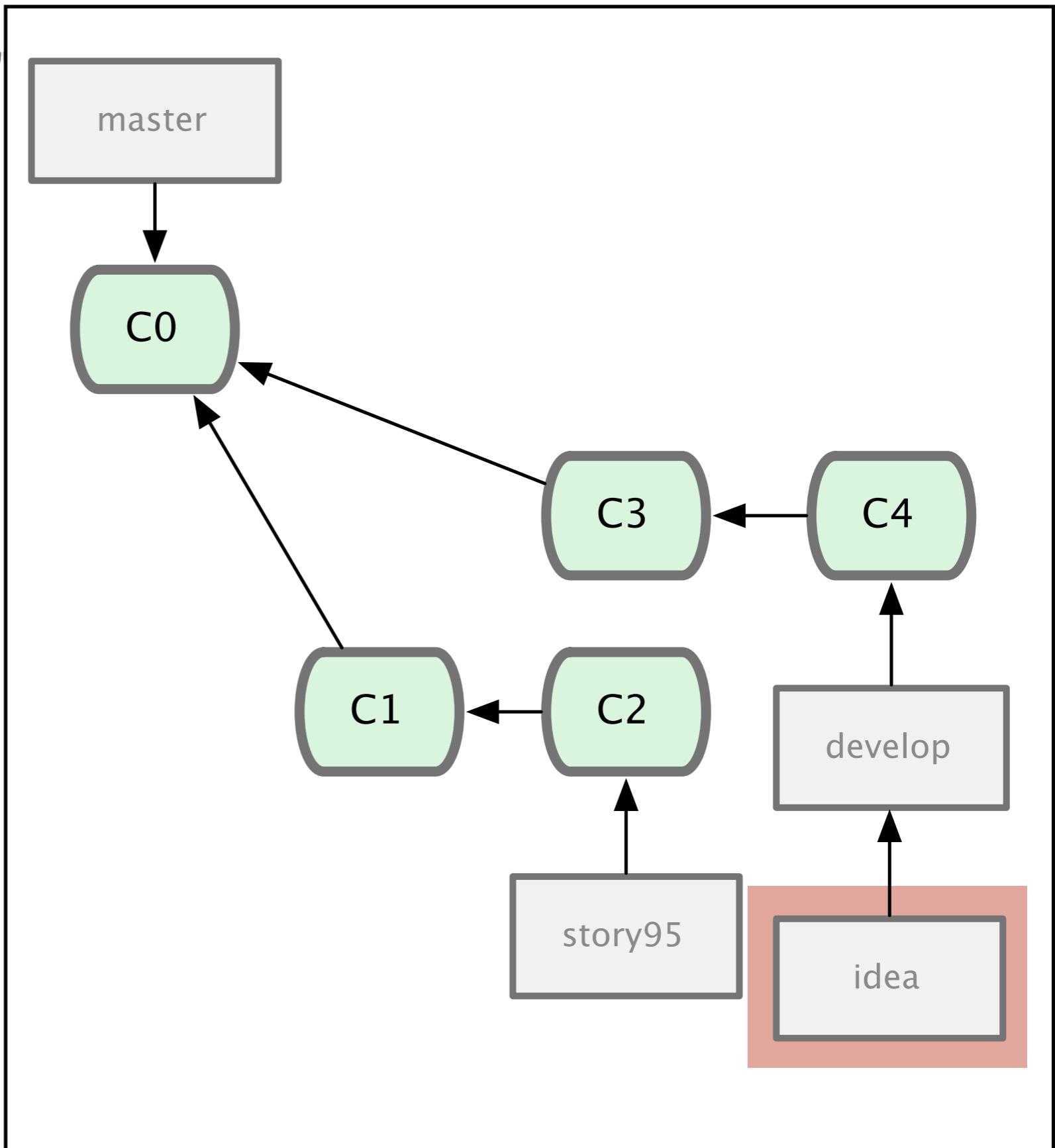
```
$>git checkout -b idea  
Switched to a new branch "idea"  
$>
```

```
$>git checkout -b idea
Switched to a new branch "idea"
$>git branch
  develop
* idea
  master
  story95
$>
```

```
$>git checkout -b idea  
Switched to a new branch  
$>git branch  
  develop  
* idea  
  master  
  story95  
$>
```



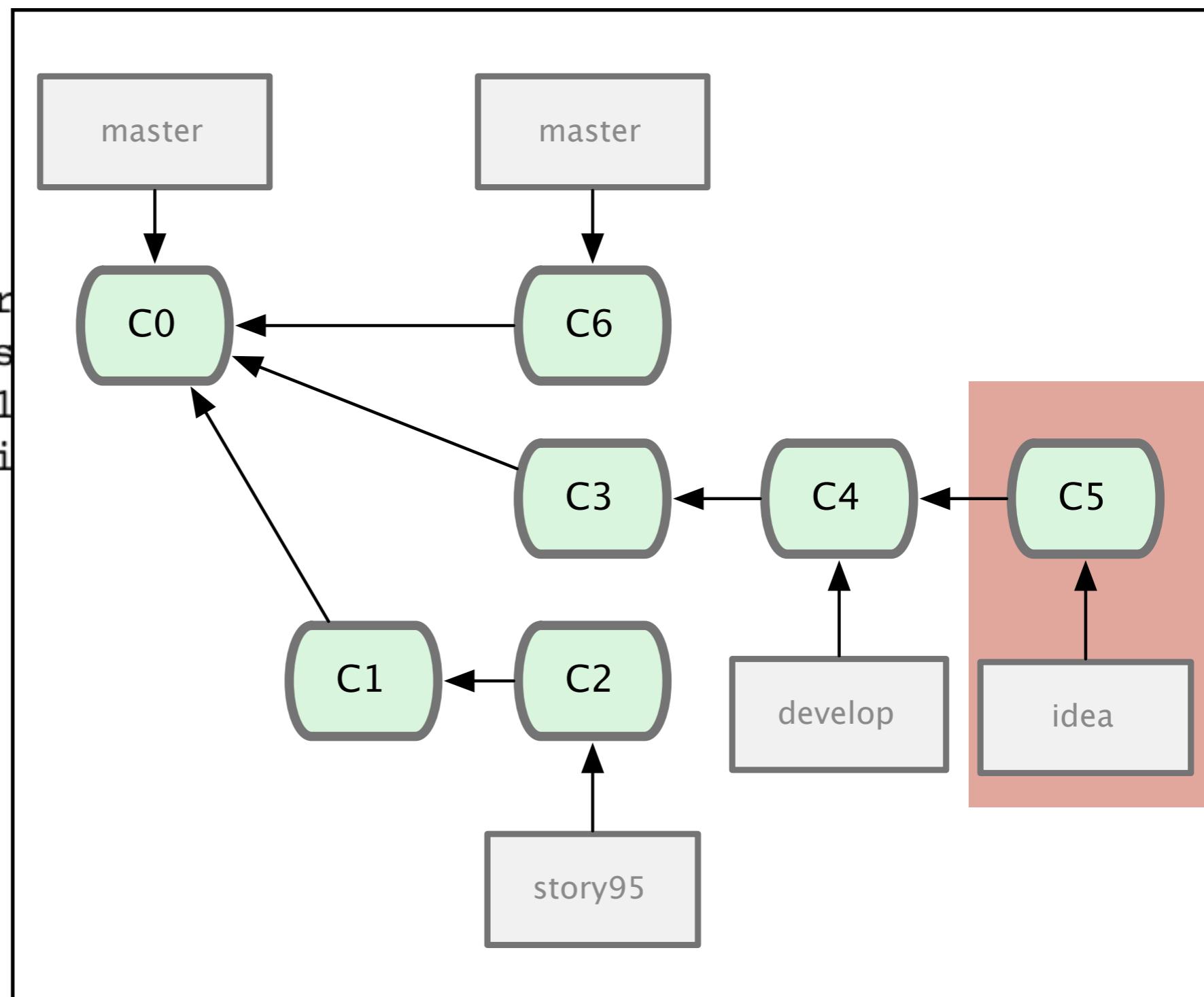
```
$>git checkout -b idea
Switched to a new branch
$>git branch
  develop
* idea
  master
  story95
$>
```



```
$>git checkout -b idea
Switched to a new branch "idea"
$>git branch
  develop
* idea
  master
  story95
$>vim lib/simplegit.rb
$>
```

```
$>git checkout -b idea
Switched to a new branch "idea"
$>git branch
  develop
* idea
  master
  story95
$>vim lib/simplegit.rb
$>git commit -a -m 'split show command into raw and normal'
Created commit 8268cla: split show command into raw and normal
  1 files changed, 4 insertions(+), 0 deletions(-)
$>
```

```
$>git checkout -b idea
Switched to a new branch "idea"
$>git branch
  develop
* idea
  master
  story95
$>vim lib/simplegit.r
$>git commit -a -m 's
Created commit 8268c1
  1 files changed, 4 i
```



branching workflow

normal development

want to try an idea

have to do a hotfix

merge idea into development

push to production

```
$>git checkout master  
Switched to branch "master"  
$>
```

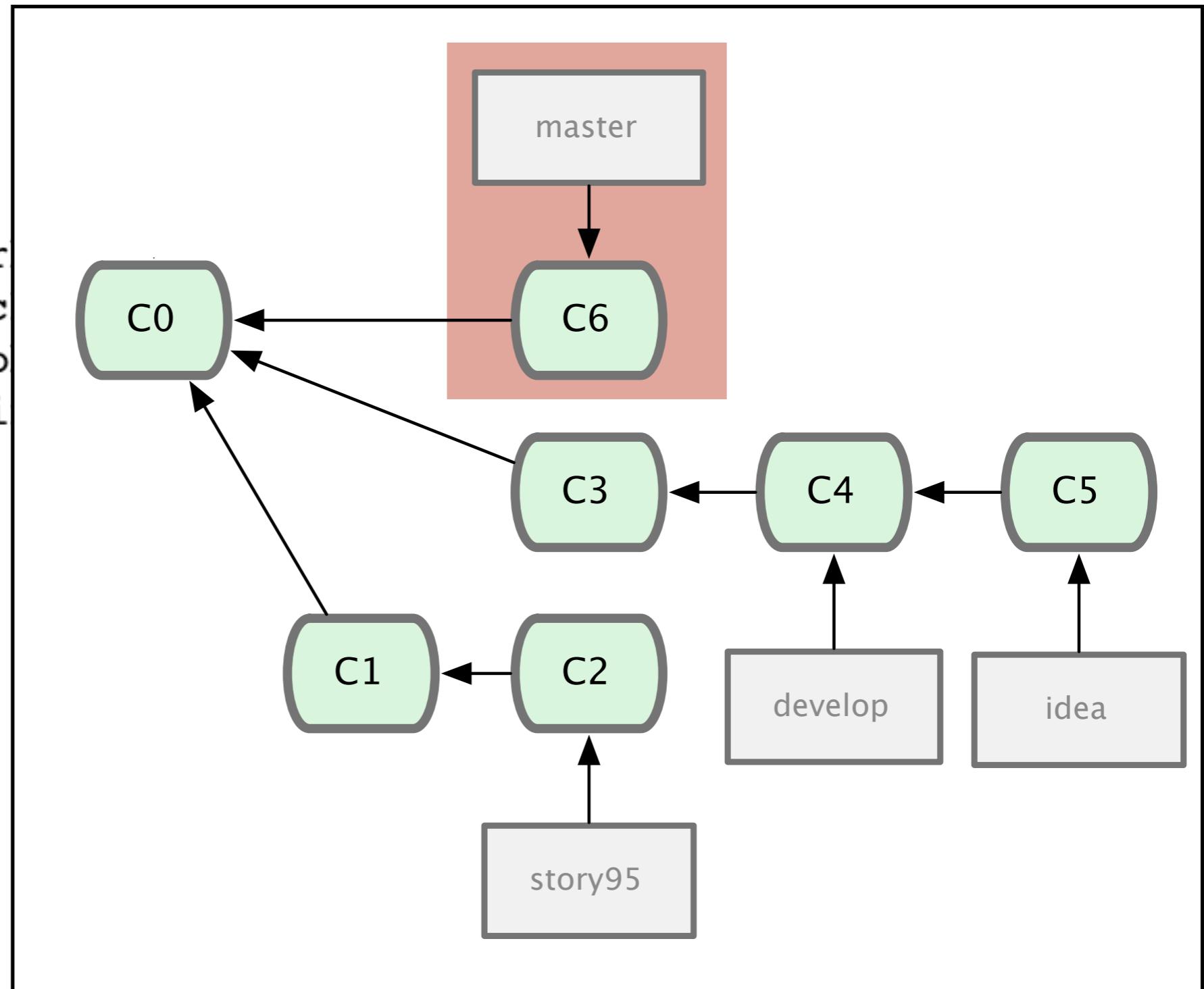
```
$>git checkout master  
Switched to branch "master"  
$>git branch  
  develop  
  idea  
* master  
  story95  
$>
```

```
$>git checkout master
Switched to branch "master"
$>git branch
  develop
  idea
* master
  story95
$>vim lib/simplegit.rb
$>
```

```
$>git checkout master
Switched to branch "master"
$>git branch
  develop
  idea
* master
  story95
$>vim lib/simplegit.rb
$>git commit -a -m 'changed log output to 30 from 25'
Created commit 749bcbb: changed log output to 30 from 25
  1 files changed, 1 insertions(+), 1 deletions(-)
$>
```

```
$>git checkout master
Switched to branch "master"
$>git branch
  develop
  idea
* master
  story95
$>vim lib/simplegit.rb
$>git commit -a -m 'changed log output to 30 from 25'
Created commit 749bcbb: changed log output to 30 from 25
  1 files changed, 1 insertions(+), 1 deletions(-)
$>git tag -a 'v1.3'
$>
```

```
$>git checkout master  
Switched to branch "master"  
$>git branch  
  develop  
  idea  
* master  
  story95  
$>vim lib/simplegit.r  
$>git commit -a -m 'c  
Created commit 749bcb  
  1 files changed, 1 i  
$>git tag -a 'v1.3'
```



branching workflow

normal development

want to try an idea

have to do a hotfix

merge idea into development

push to production

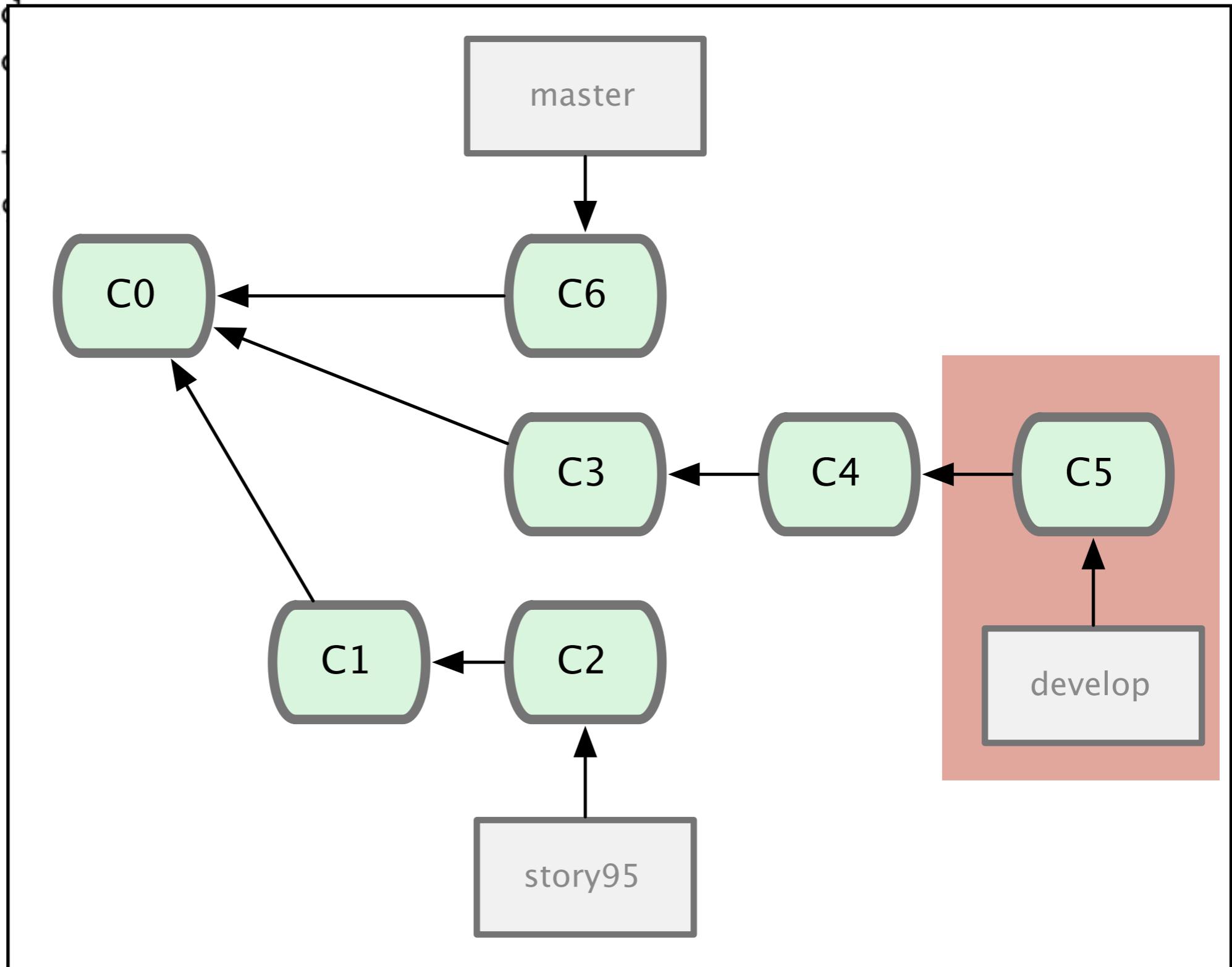
```
$>git checkout develop
```

```
$>git checkout develop  
Switched to branch "develop"  
$>
```

```
$>git checkout develop
Switched to branch "develop"
$>git merge idea
Updating 8f35c23..8268c1a
Fast forward
  lib/simplegit.rb |      4 +++
  1 files changed, 4 insertions(+), 0 deletions(-)
$>
```

```
$>git checkout develop
Switched to branch "develop"
$>git merge idea
Updating 8f35c23..8268c1a
Fast forward
  lib/simplegit.rb |      4 +++
  1 files changed, 4 insertions(+), 0 deletions(-)
$>
```

```
$>git checkout develop
Switched to branch "develop"
$>git merge id
Updating 8f35c...
Fast forward
 lib/simplegi...
1 files changed
```



branching workflow

normal development

want to try an idea

have to do a hotfix

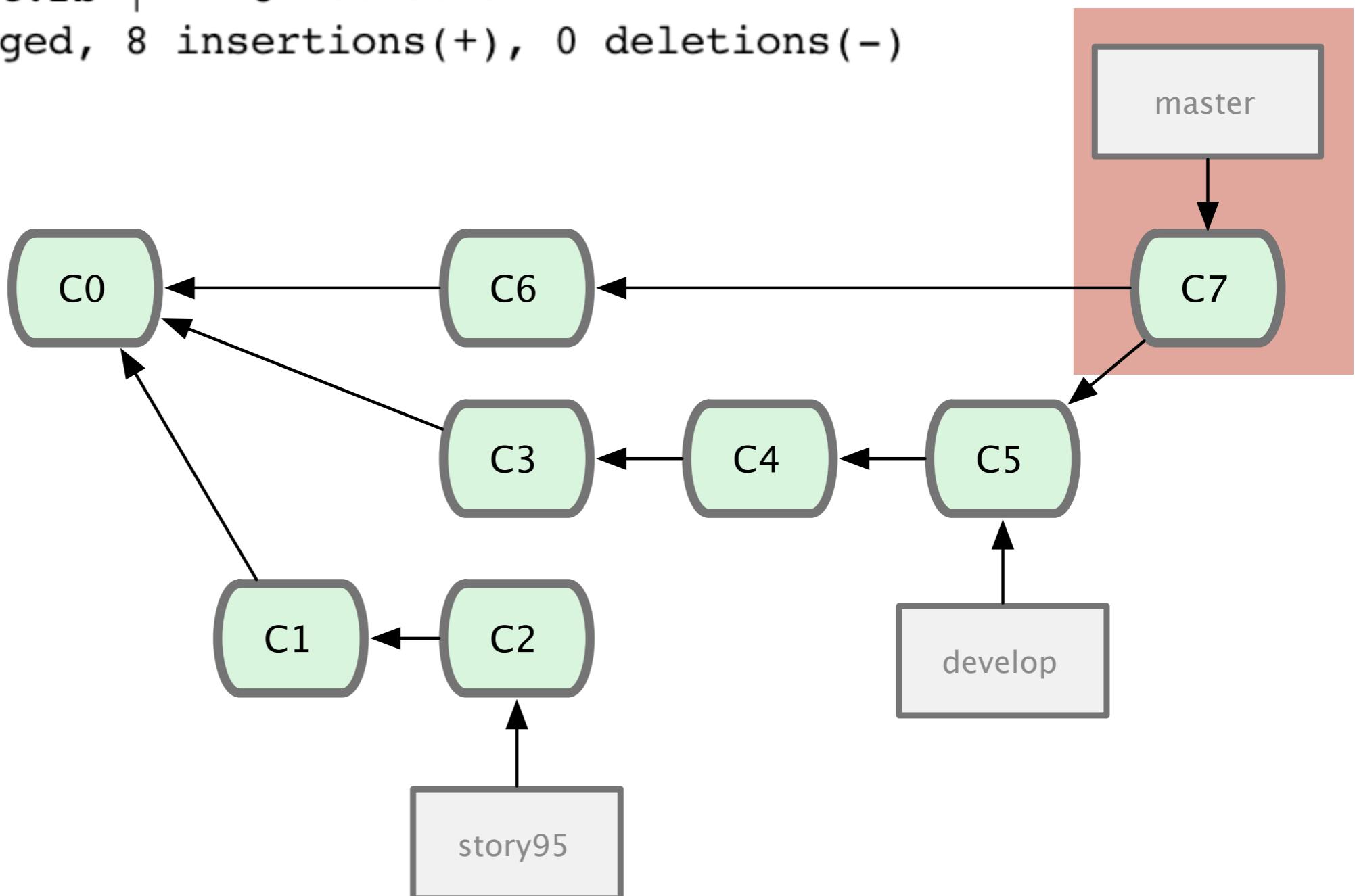
merge idea into development

push to production

```
$>git checkout master  
Switched to branch "master"  
$>
```

```
$>git checkout master
Switched to branch "master"
$>git merge develop
Auto-merged lib/simplegit.rb
Merge made by recursive.
  lib/simplegit.rb |    8 ++++++++
  1 files changed, 8 insertions(+), 0 deletions(-)
$>
```

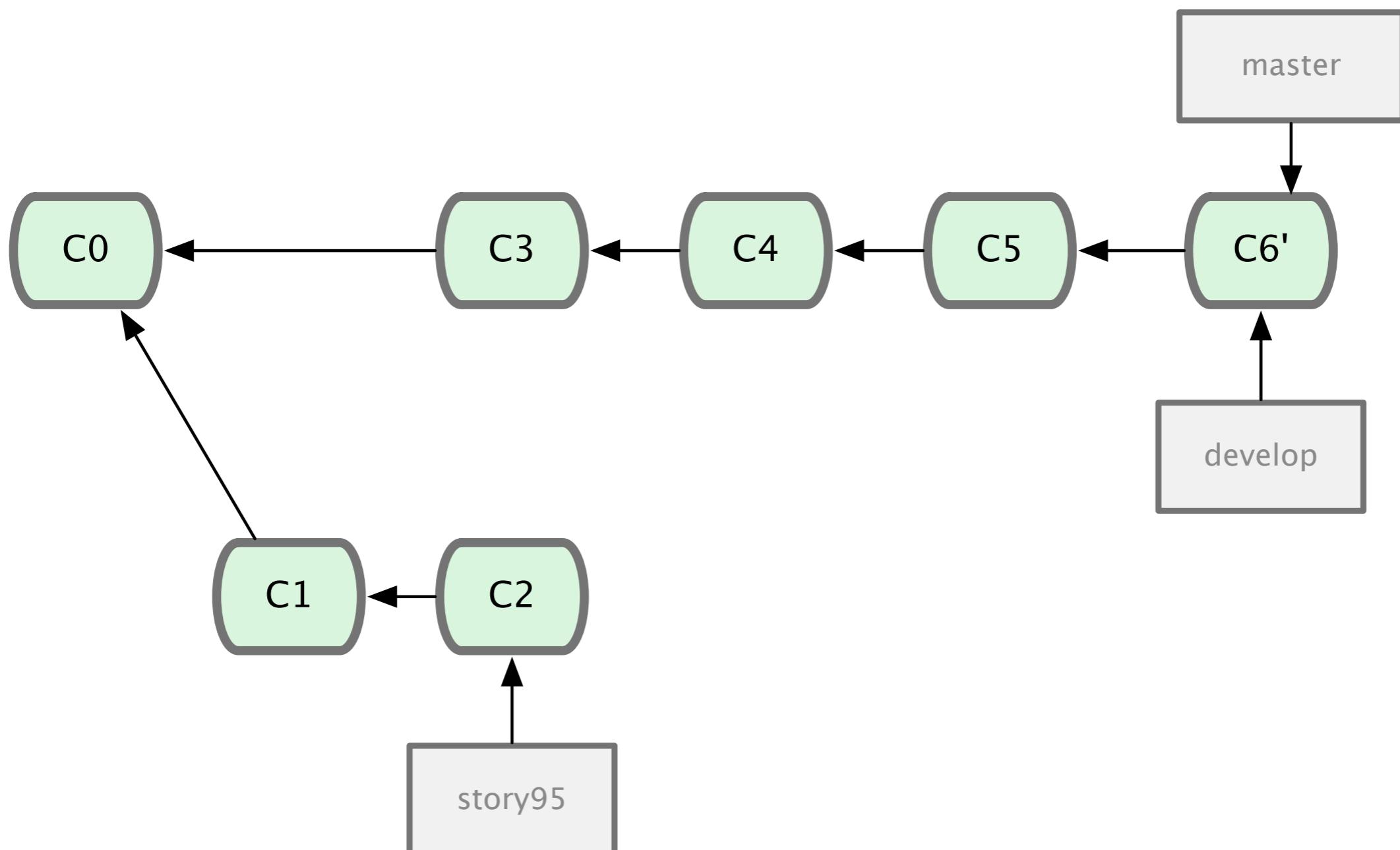
```
$>git checkout master
Switched to branch "master"
$>git merge develop
Auto-merged lib/simplegit.rb
Merge made by recursive.
  lib/simplegit.rb |    8 ++++++++
  1 files changed, 8 insertions(+), 0 deletions(-)
$>
```



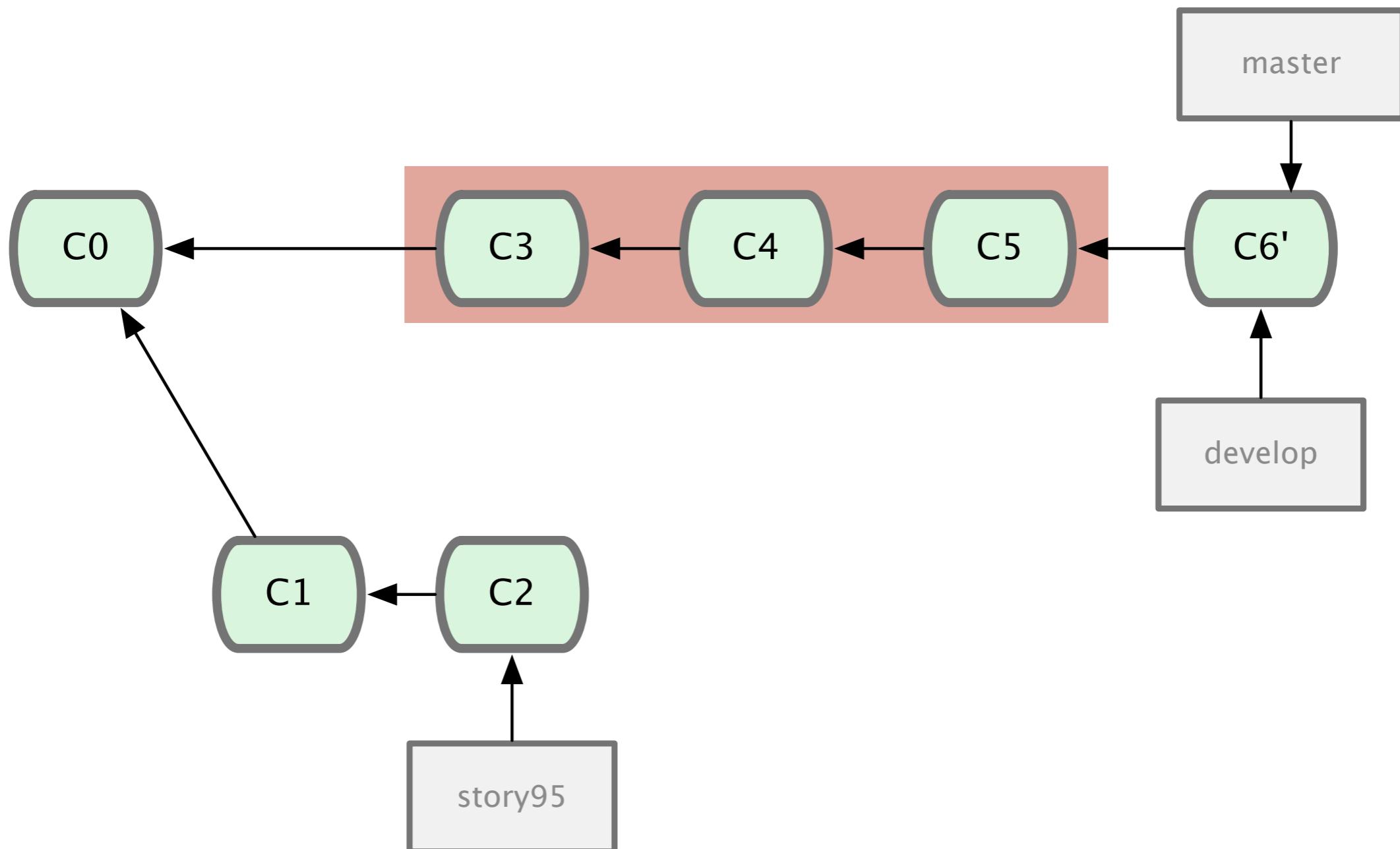
```
$>git checkout master  
Switched to branch "master"  
$>
```

```
$>git checkout master
Switched to branch "master"
$>git rebase develop
First, rewinding head to replay your work on top of it...
HEAD is now at 8268cla split show command into raw and normal
Applying changed log output to 30 from 25
$>
```

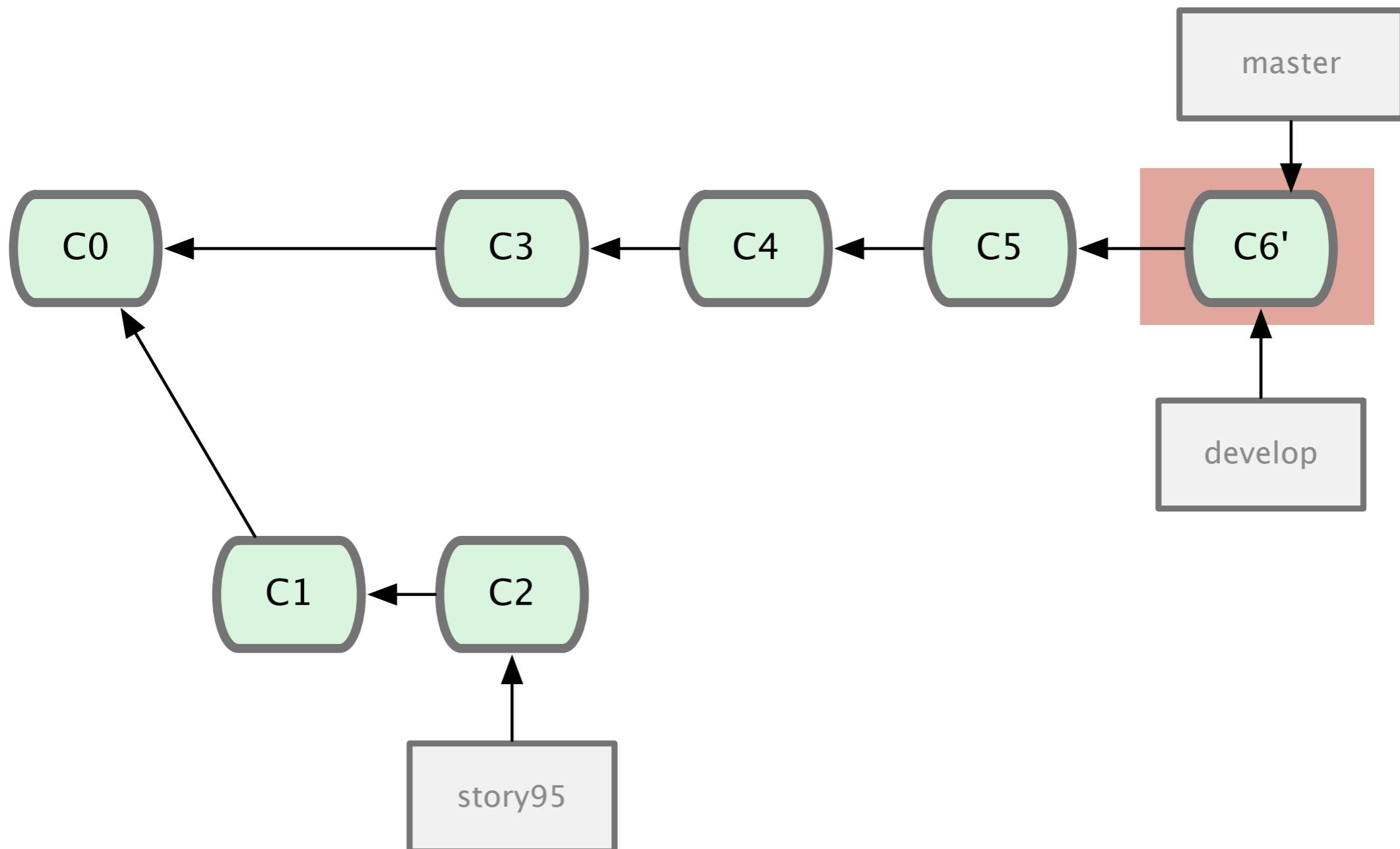
```
$>git checkout master
Switched to branch "master"
$>git rebase develop
First, rewinding head to replay your work on top of it...
HEAD is now at 8268cla split show command into raw and normal
Applying changed log output to 30 from 25
$>
```



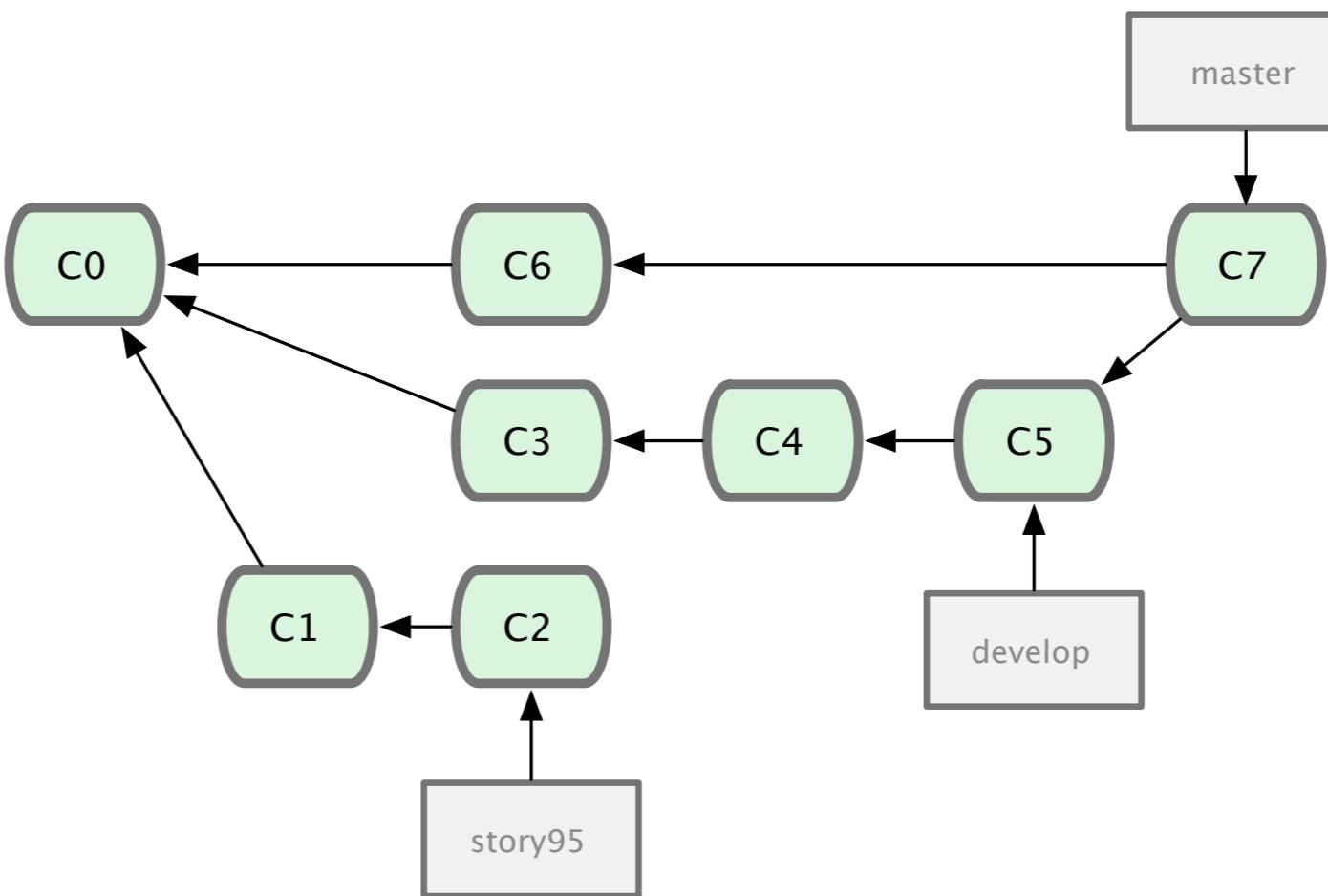
```
$>git checkout master
Switched to branch "master"
$>git rebase develop
First, rewinding head to replay your work on top of it...
HEAD is now at 8268cla split show command into raw and normal
Applying changed log output to 30 from 25
$>
```



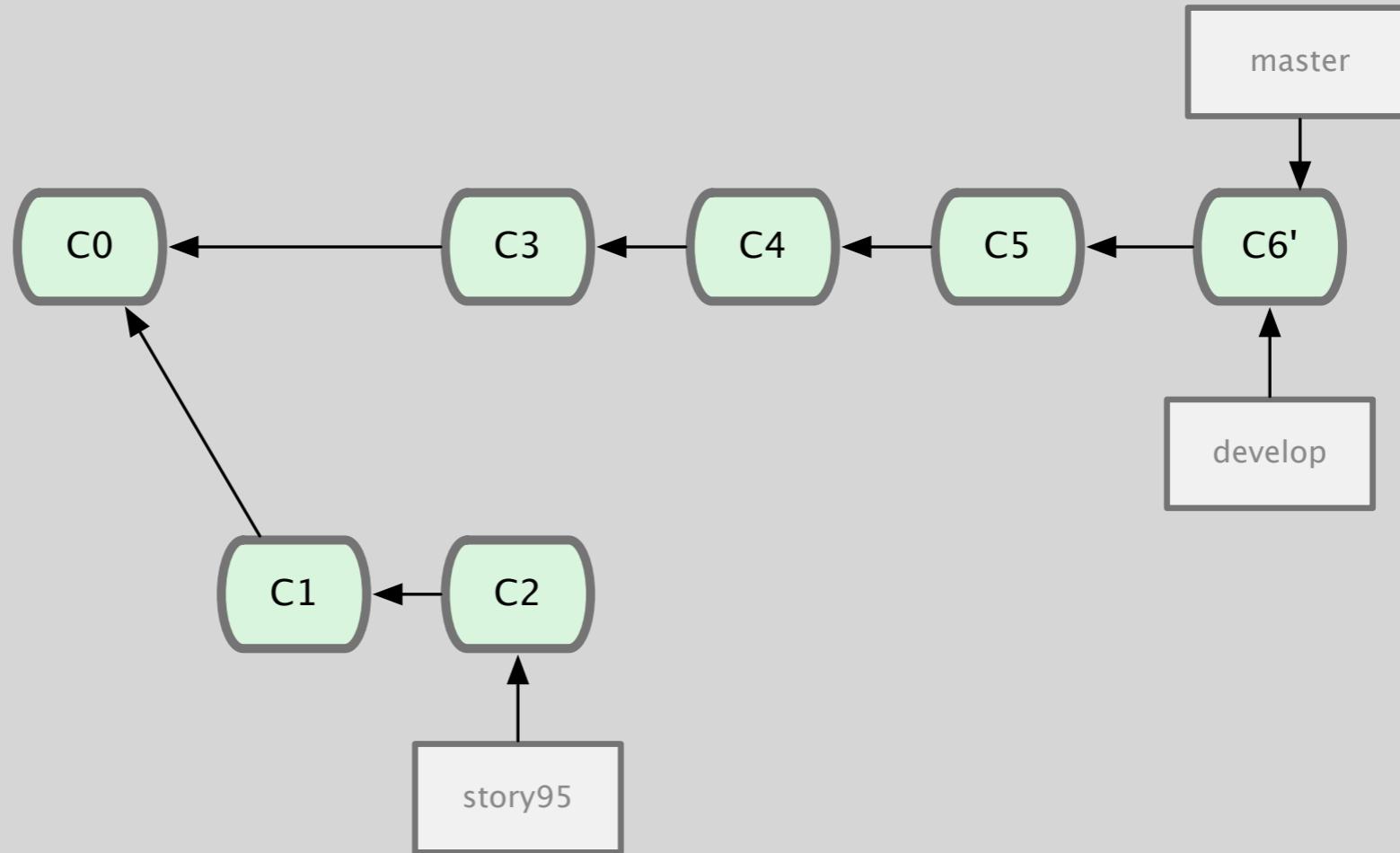
```
$>git checkout master  
Switched to branch "master"  
$>git rebase develop  
First, rewinding head to replay your work on top of it...  
HEAD is now at 8268cla split show command into raw and normal  
Applying changed log output to 30 from 25  
$>
```



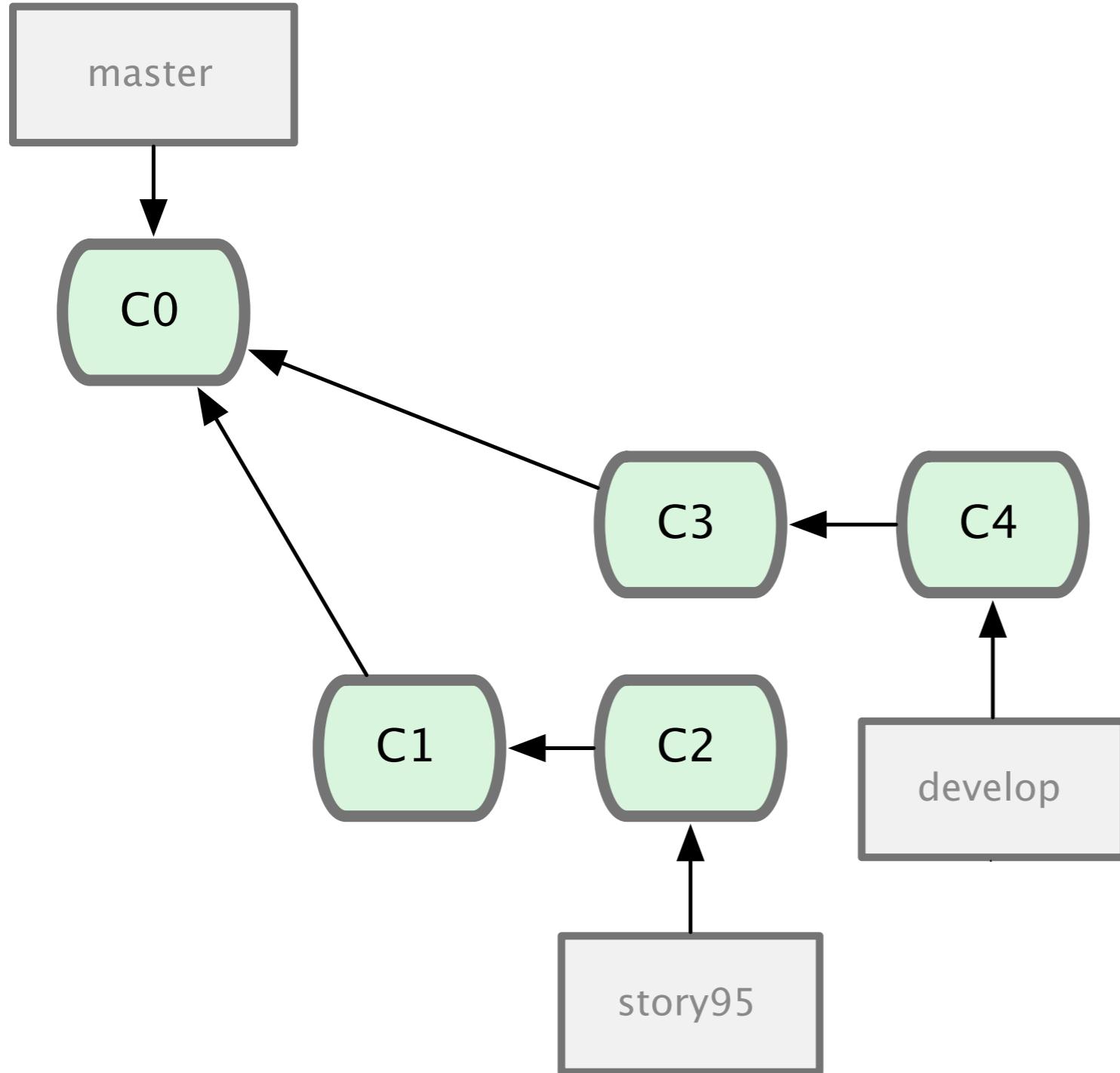
merge

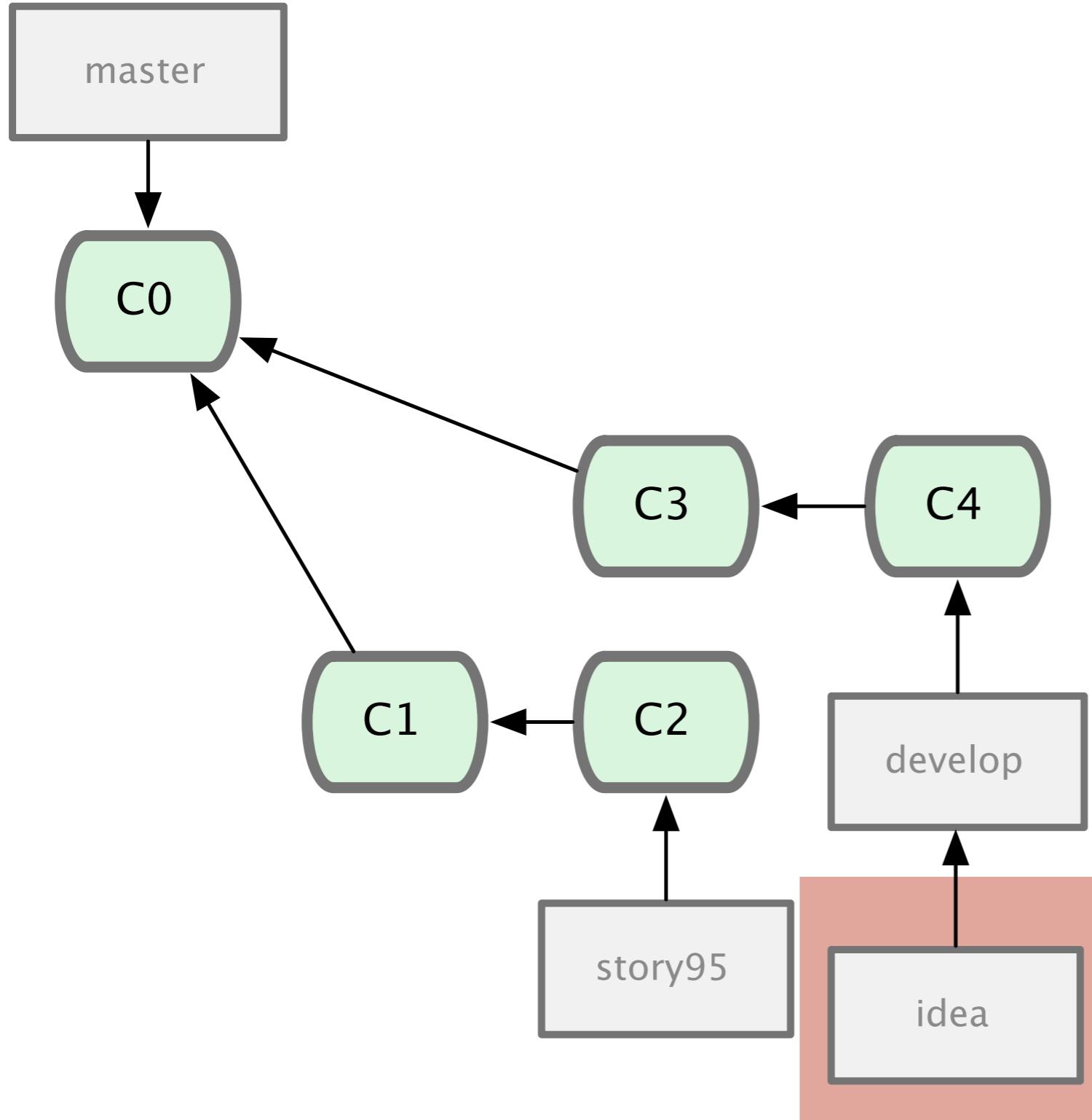


rebase

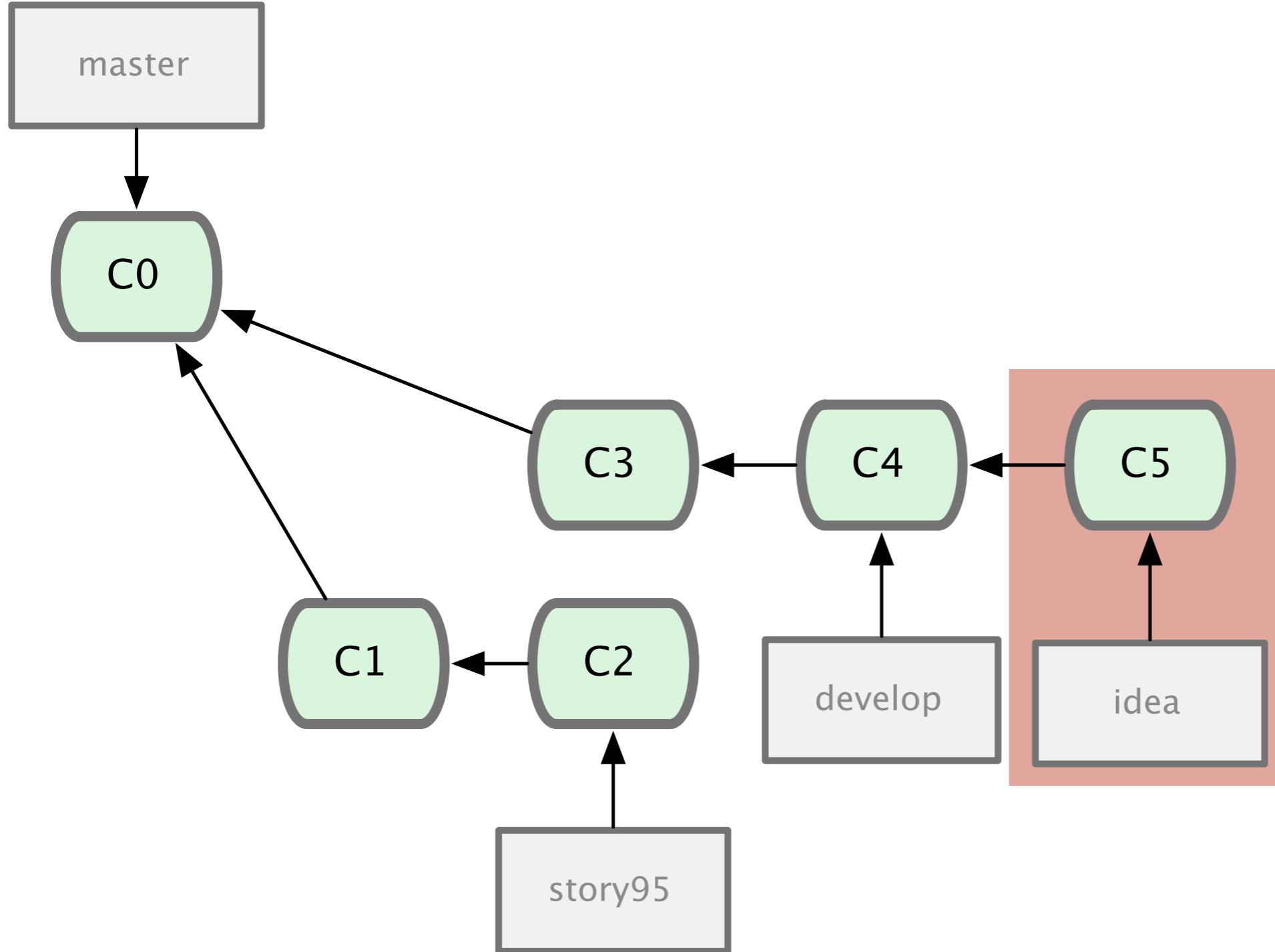


one more time now!

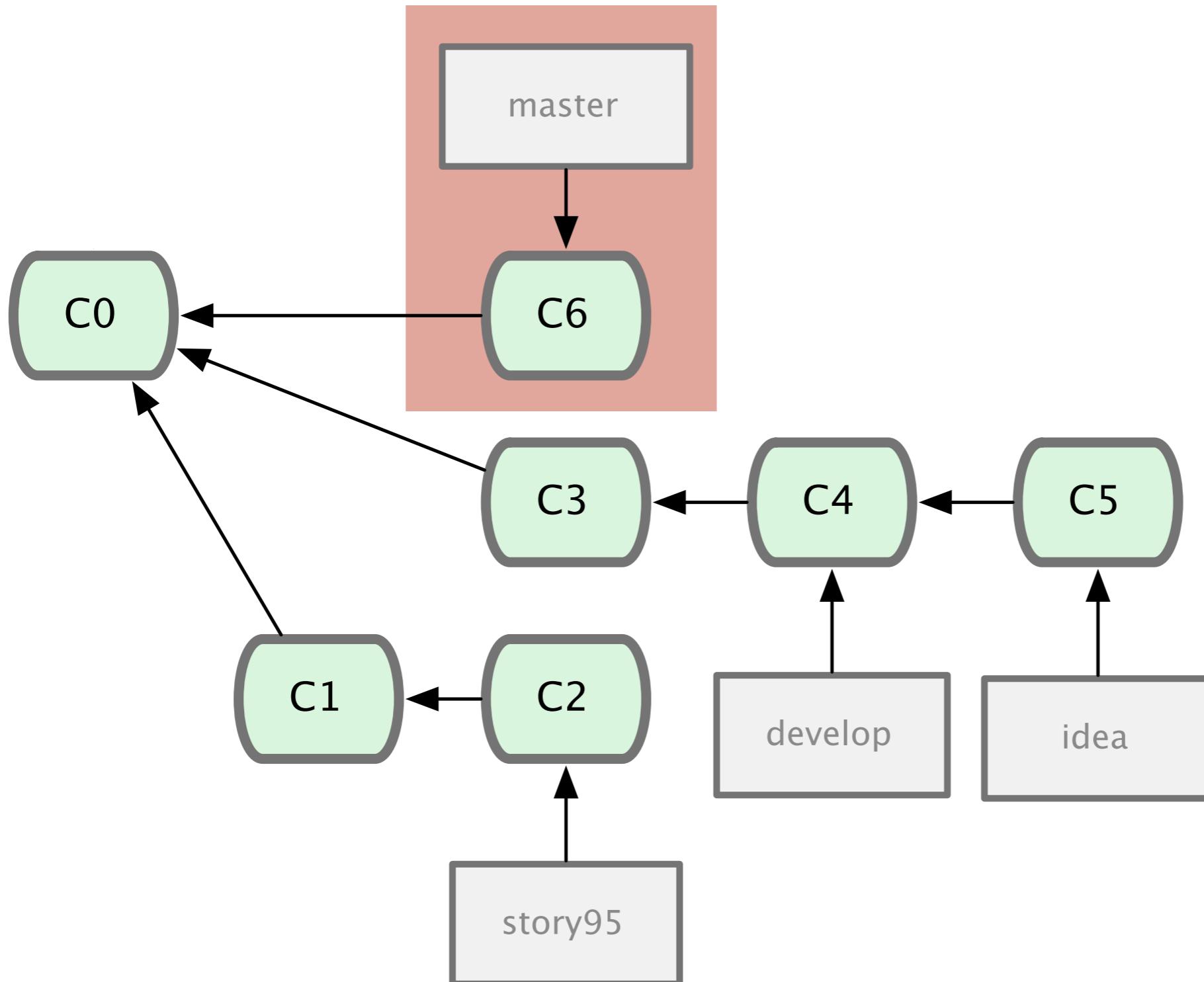




git checkout -b idea

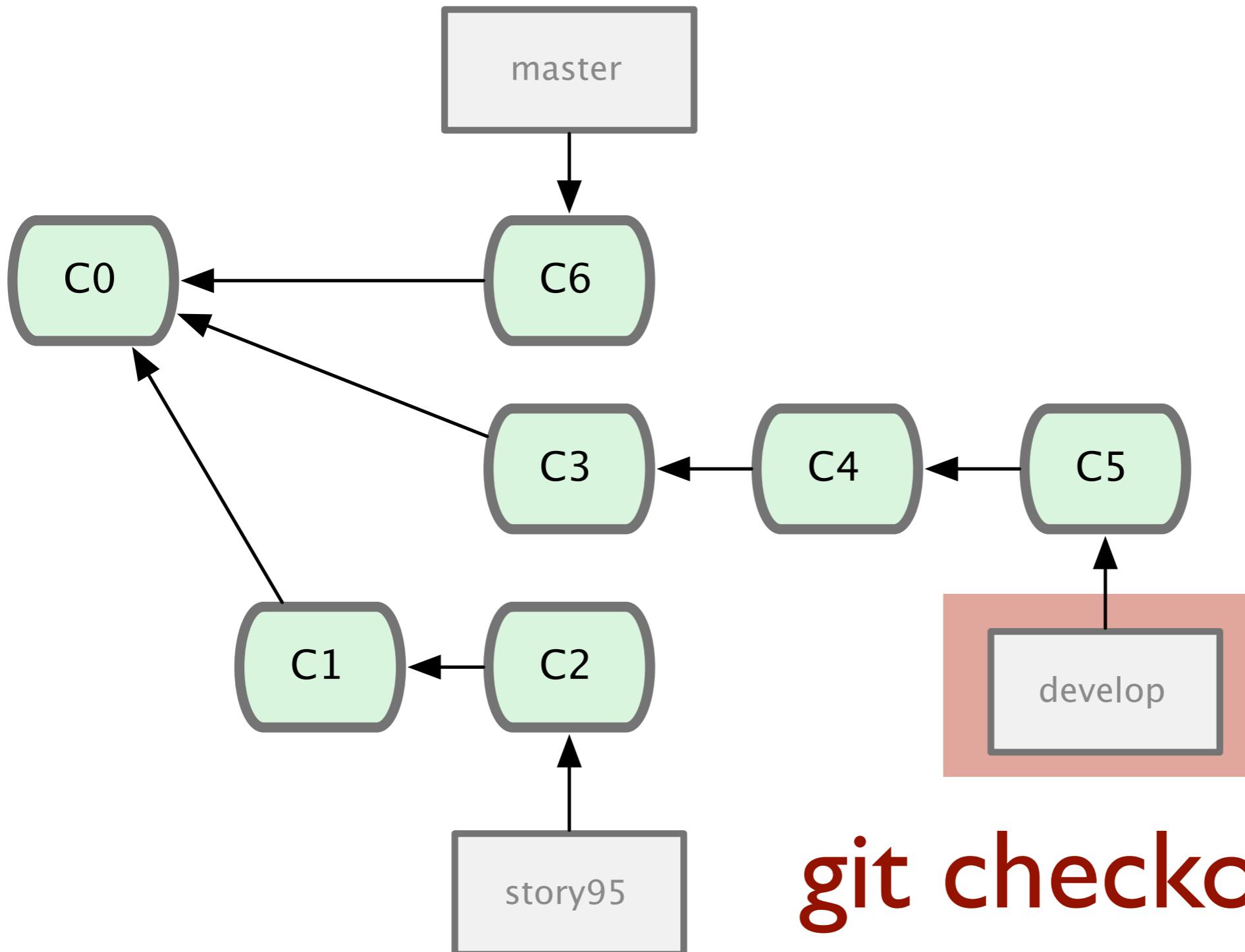


git commit

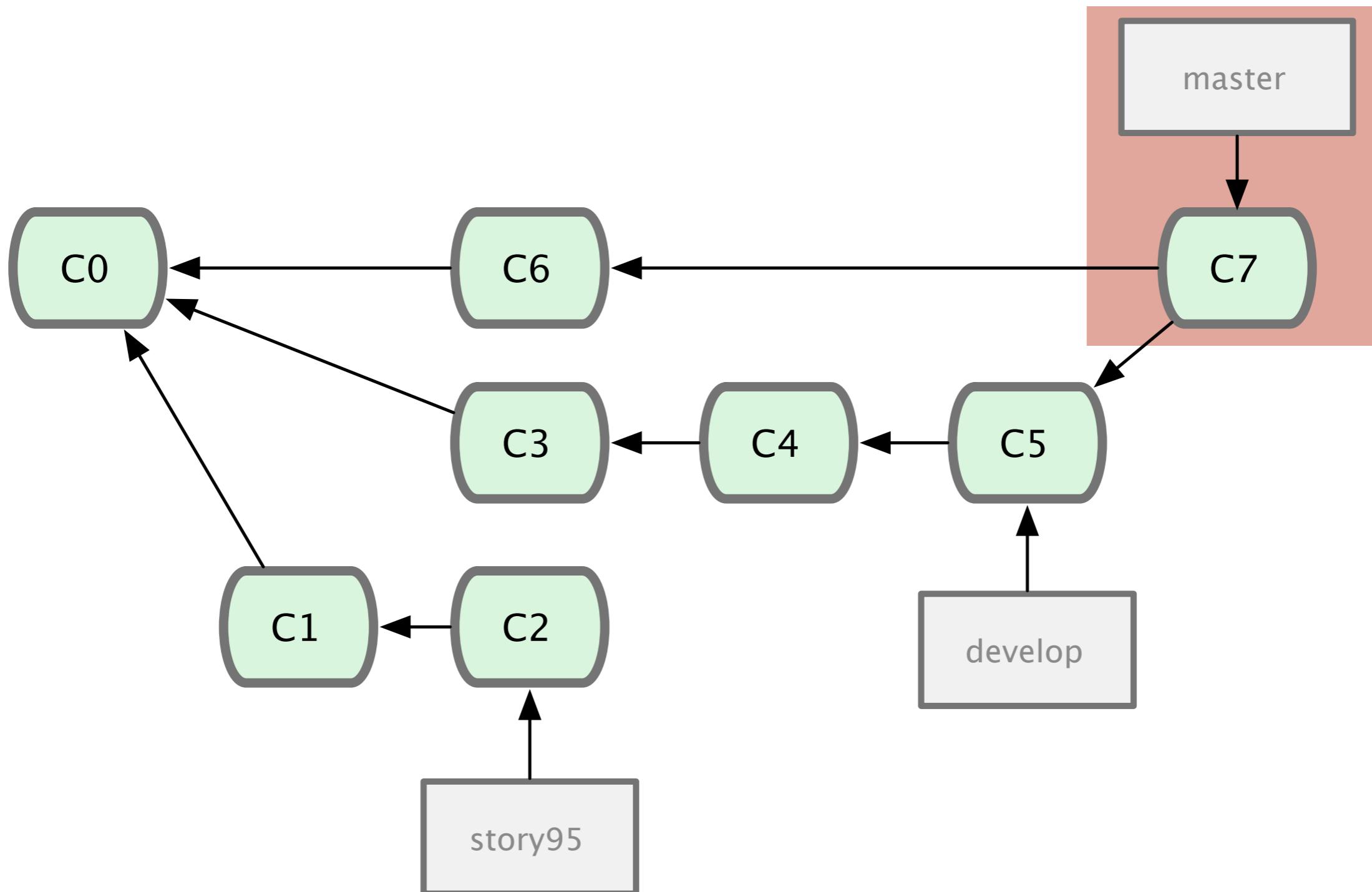


git checkout master

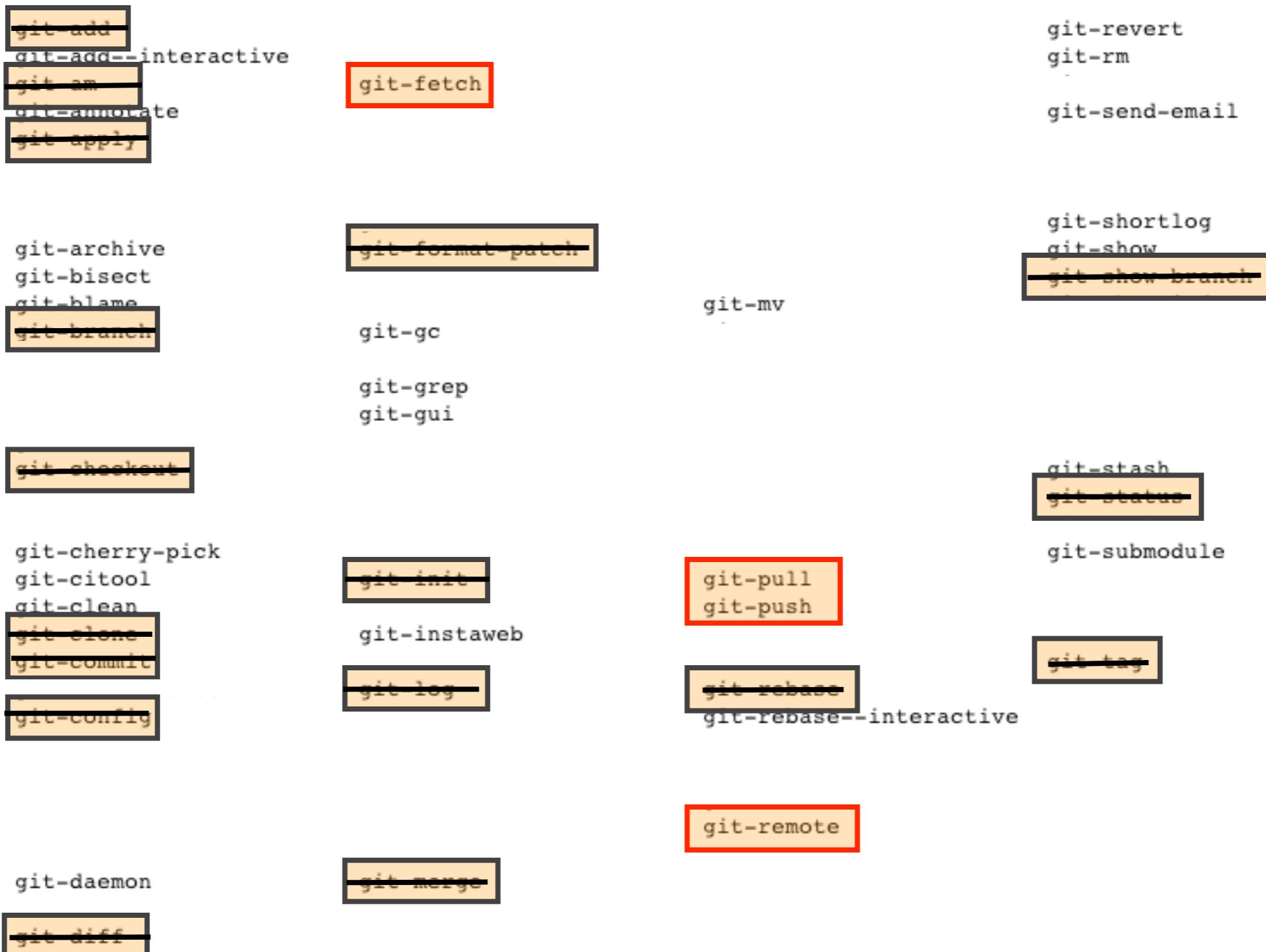
git commit



git checkout idea
git merge develop
git branch -d idea



git checkout master
git merge develop



sharing git

```
$>git remote
```

```
$>
```



schacon
[account](#) | [profile](#) | [guides](#) | [log out](#)
[✉ 0](#)

[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
[account](#) | [profile](#) | [guides](#) | [log out](#)
[✉ 0](#)

[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](#))

**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
$>_
```



schacon

[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 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

[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 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  
$>
```

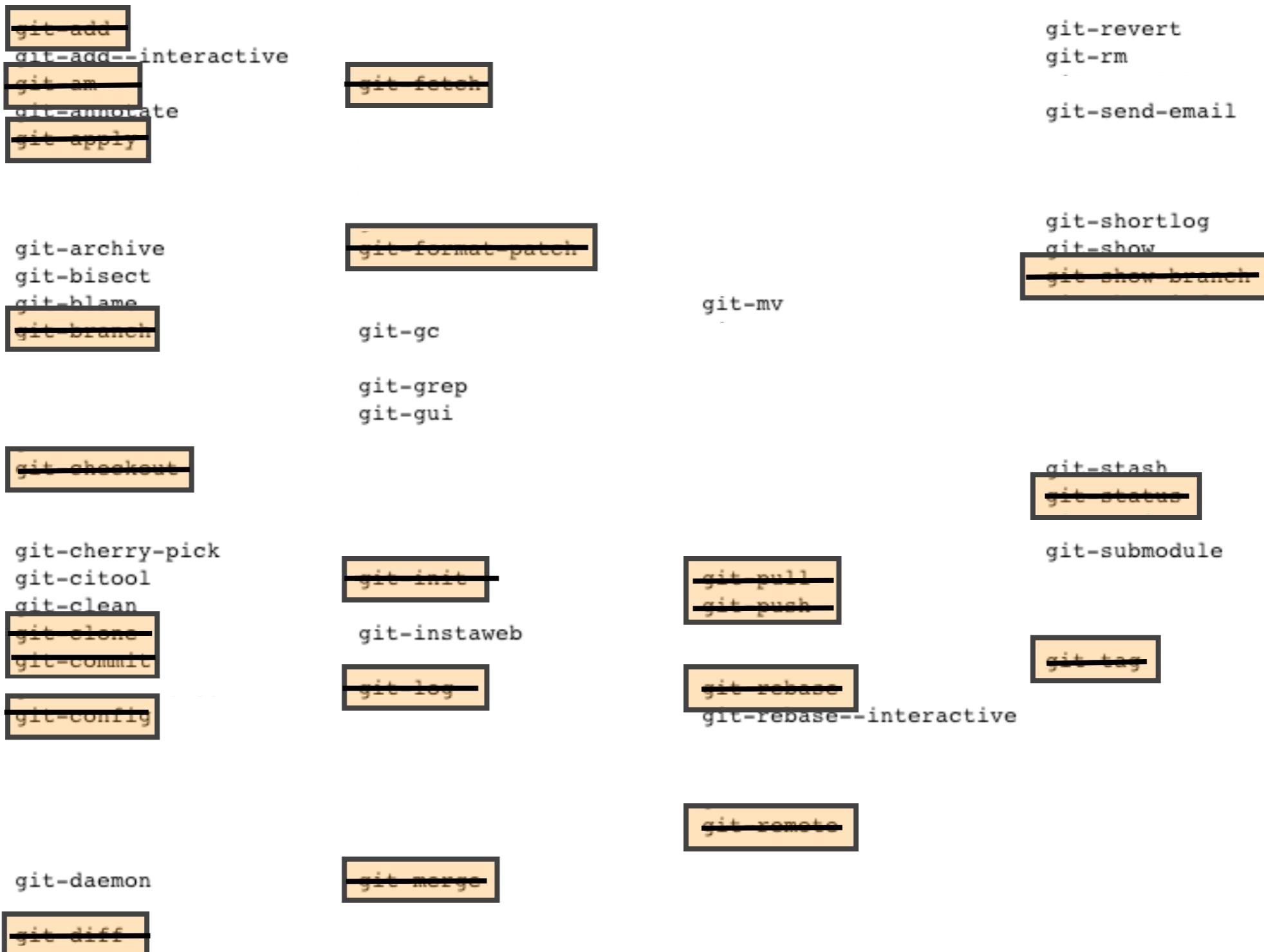
```
$>git remote  
public  
$>git remote show public  
* remote public  
  URL: git@github.com:schacon/simplegit.git  
  Tracked remote branches  
    master  
$>
```

```
$>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

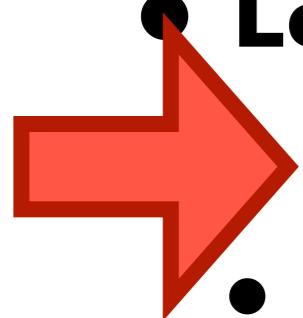
```
$>git remote  
public  
$>git remote show public  
* remote public  
  URL: git@github.com:schacon/simplegit.git  
  Tracked remote branches  
    master  
$>git push public master  
Counting objects: 7, done.  
Compressing objects: 100% (3/3), done.  
Writing objects: 100% (4/4), 365 bytes, done.  
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  
$>_
```



We Know Em All!

review

● Local Commands



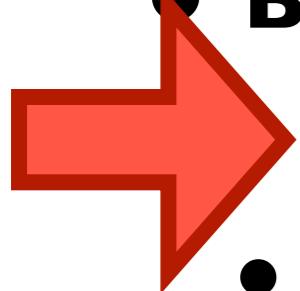
git config

- git init
- git add
- git commit
- git status
- git tag
- git log

- **Local Commands**

- git config
- git init
- git add
- git commit
- git status
- git tag
- git log

- **Branchy Commands**



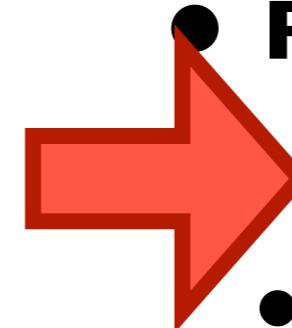
- git checkout
- git branch
- git merge
- git rebase

● Local Commands

- git config
- git init
- git add
- git commit
- git status
- git tag
- git log

● Remotey Commands

- git remote
- git fetch
- git pull
- git clone
- git push



● Branchy Commands

- git checkout
- git branch
- git merge
- git rebase

● Local Commands

- git config
- git init
- git add
- git commit
- git status
- git tag
- git log

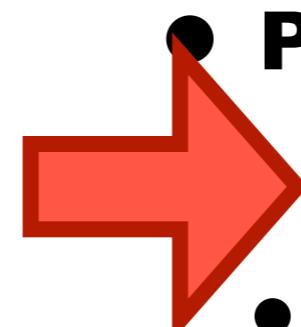
● Branchy Commands

- git checkout
- git branch
- git merge
- git rebase

● Remotey Commands

- git remote
- git fetch
- git pull
- git clone
- git push

● Patchy Commands

- 
- git diff
 - git apply
 - git format-patch
 - git am

● Local Commands

- git config
- git init
- git add
- git commit
- git status
- git tag
- git log

● Branchy Commands

- git checkout
- git branch
- git merge
- git rebase

● Remotey Commands

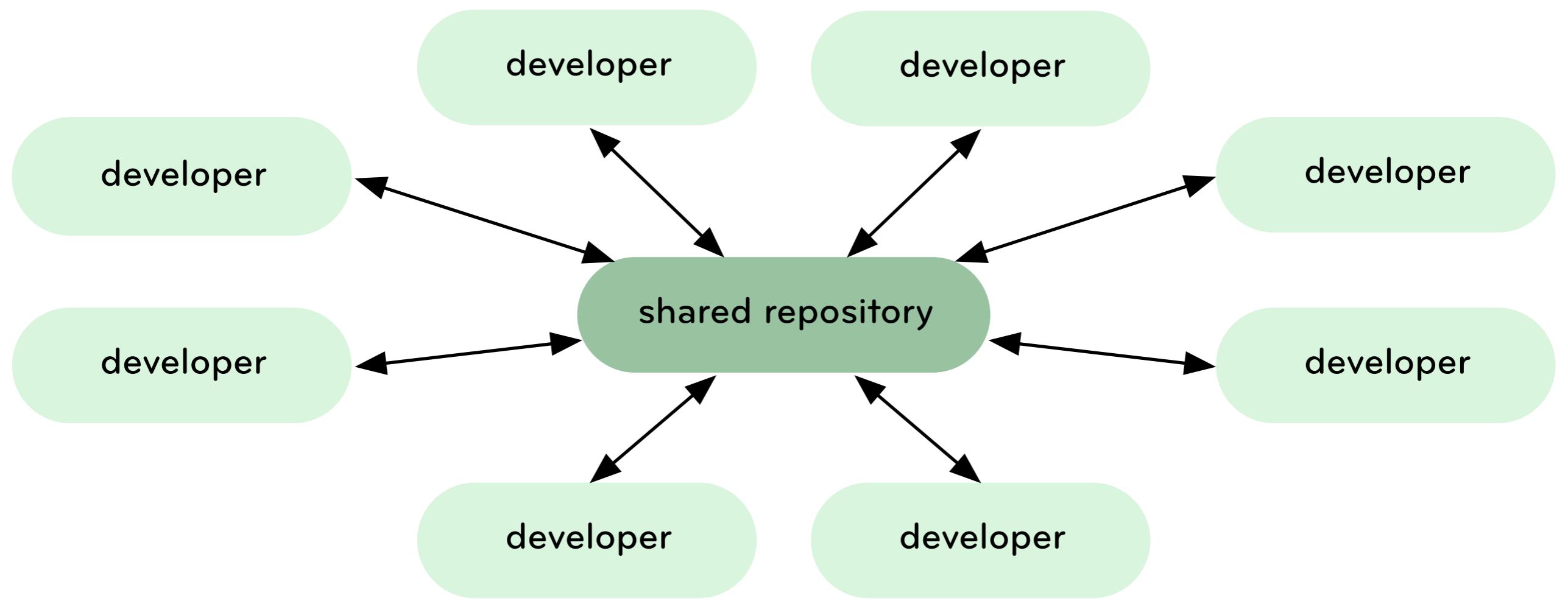
- git remote
- git fetch
- git pull
- git clone
- git push

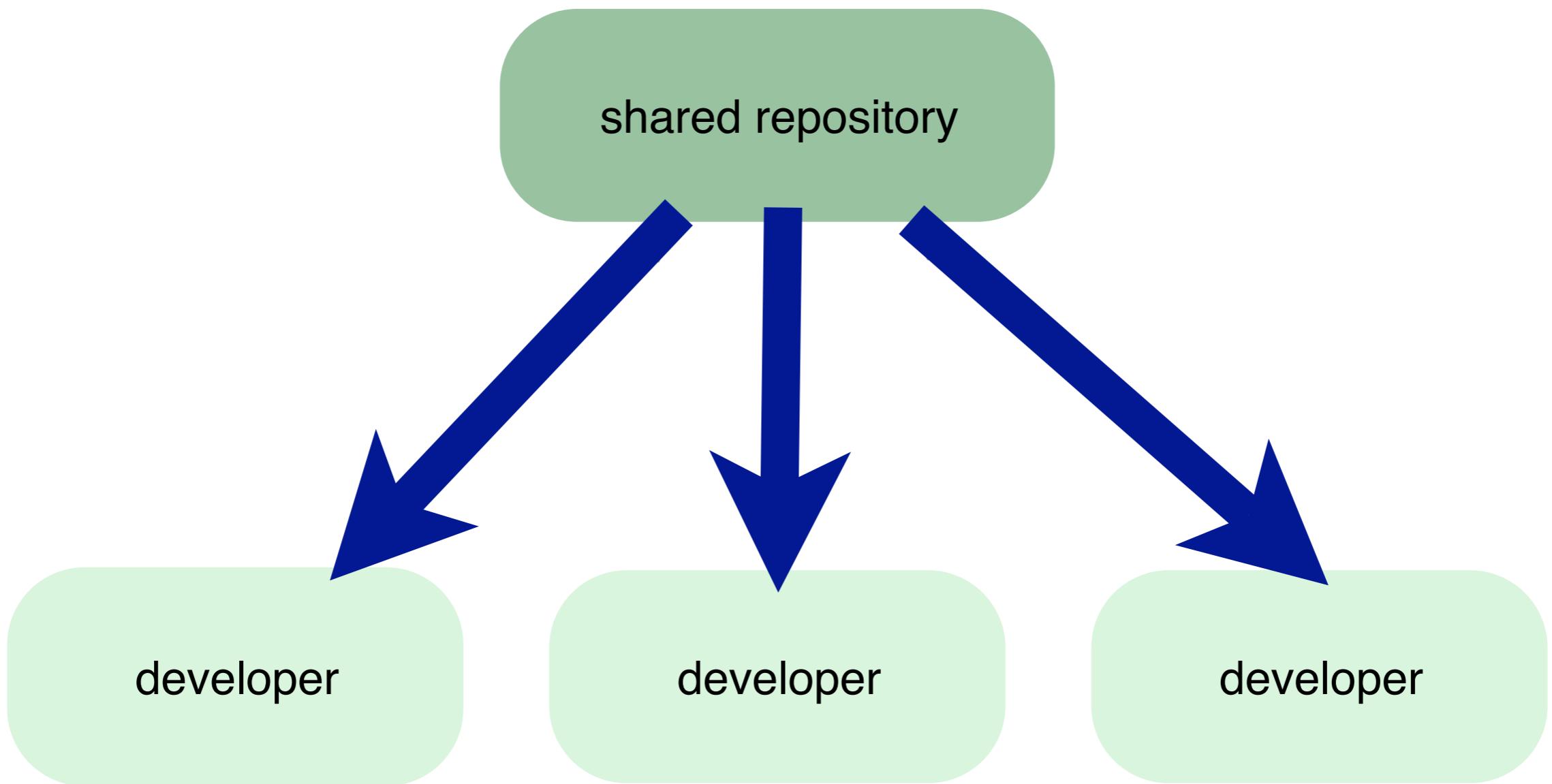
● Patchy Commands

- git diff
- git apply
- git format-patch
- git am

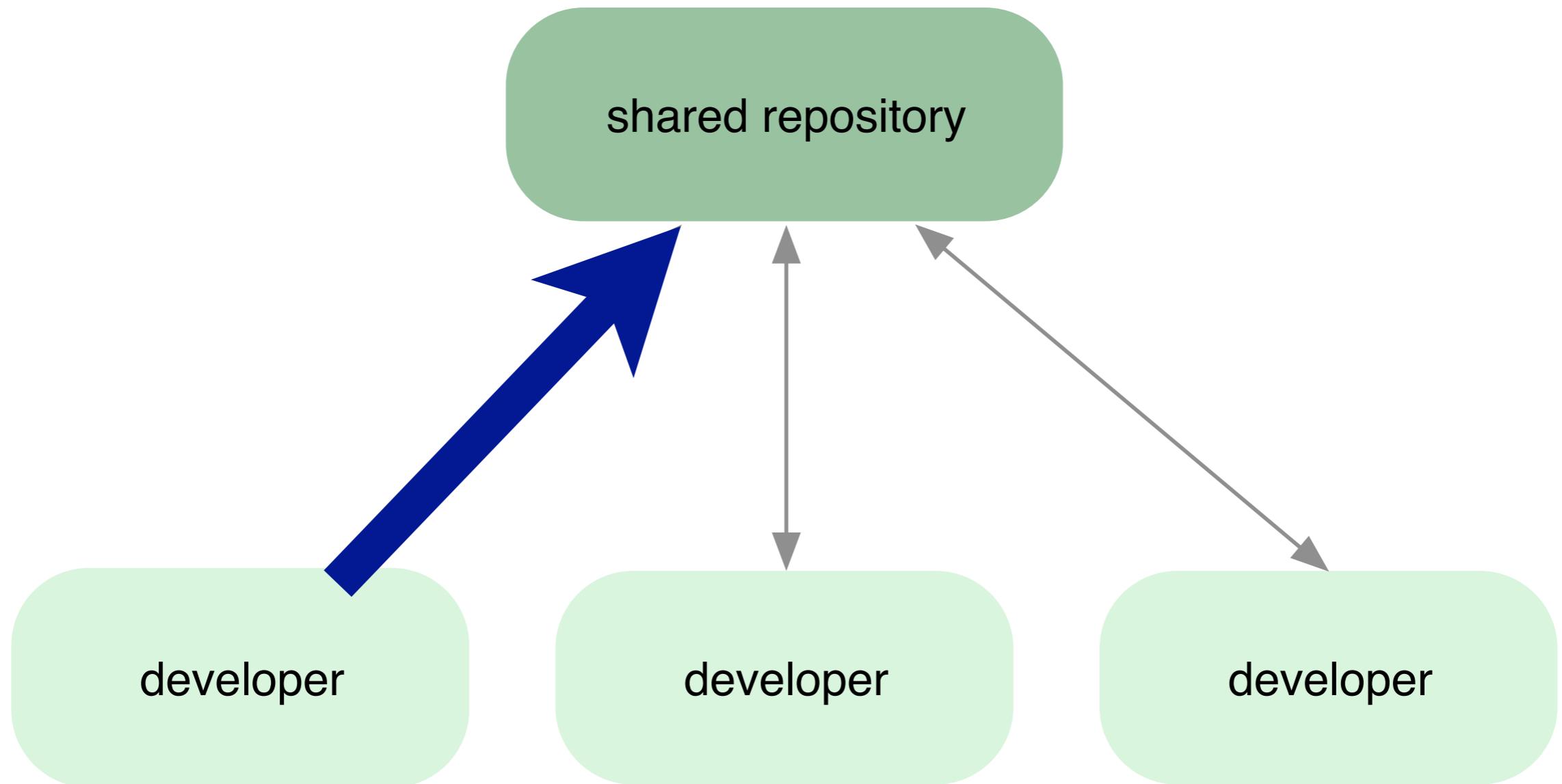
popular workflows

**central repository
model**

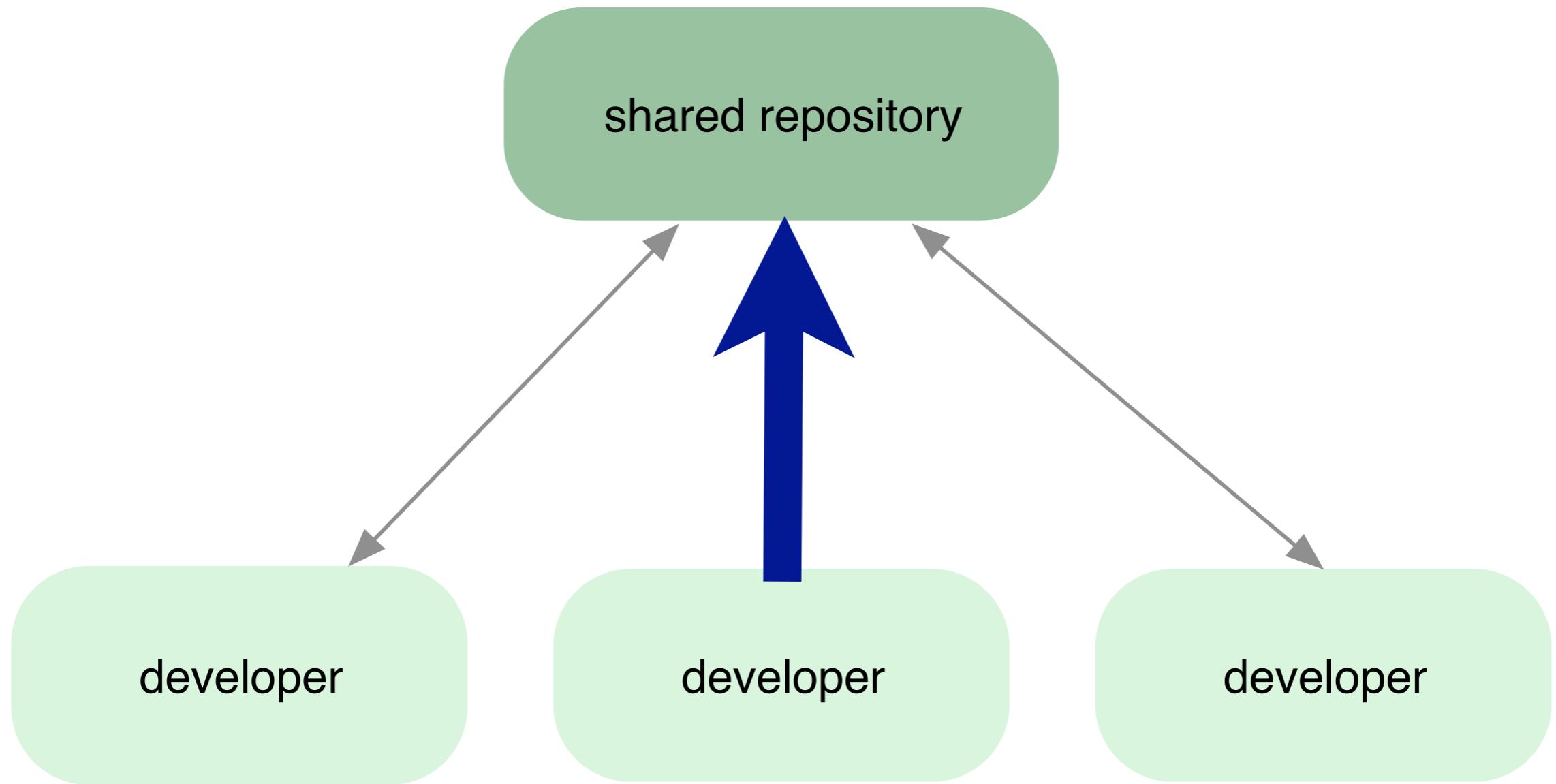




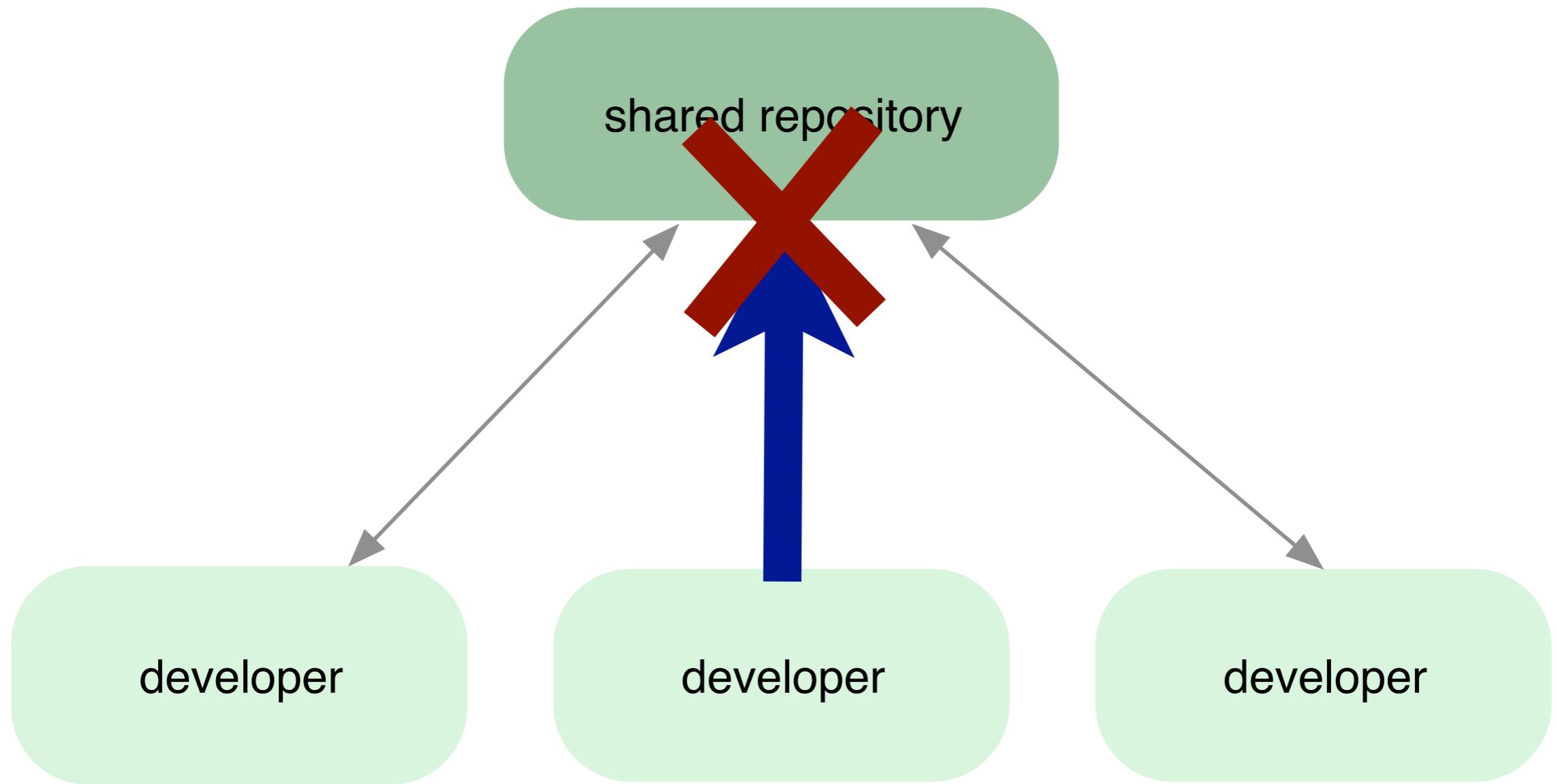
git clone

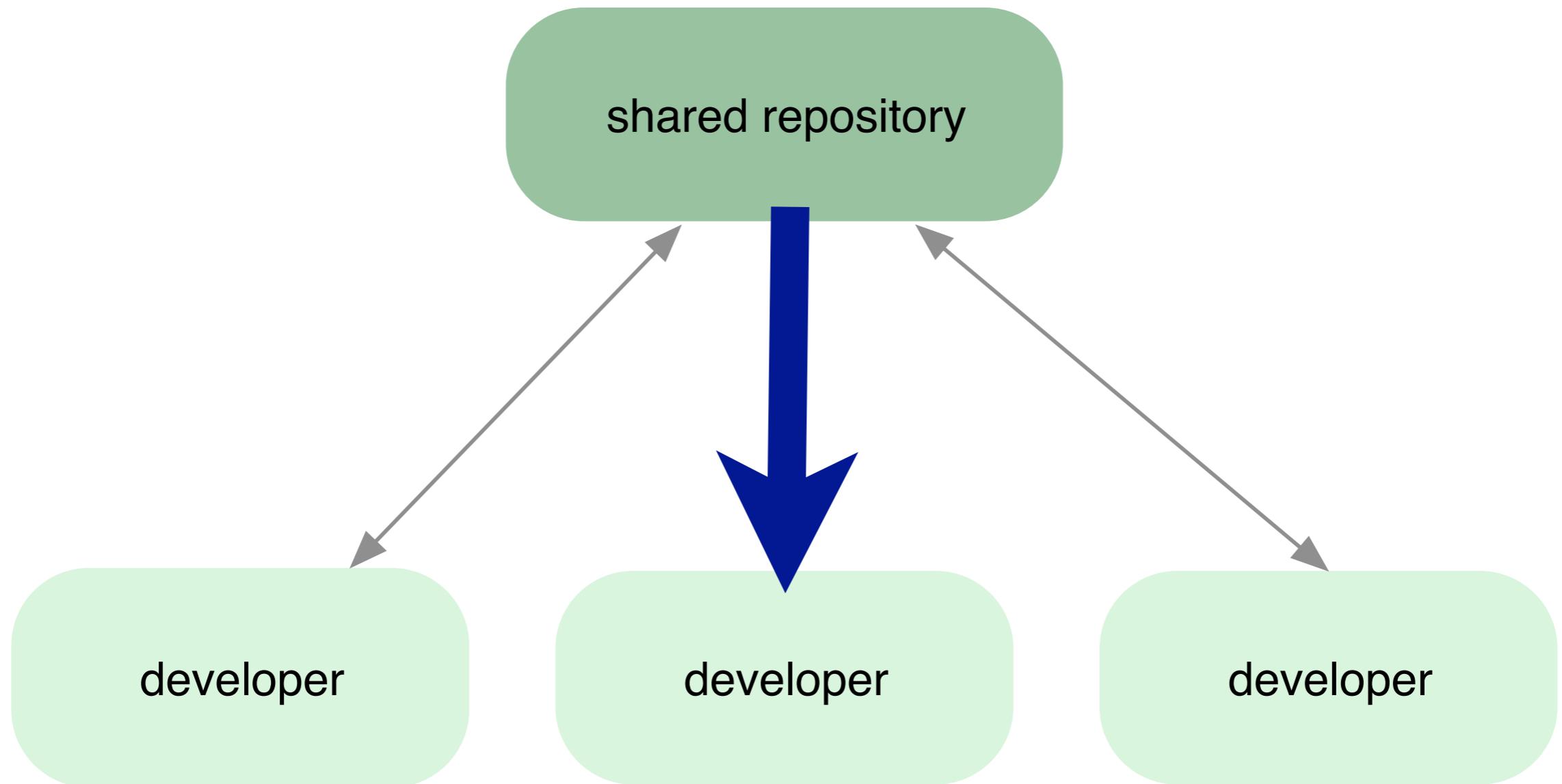


git push

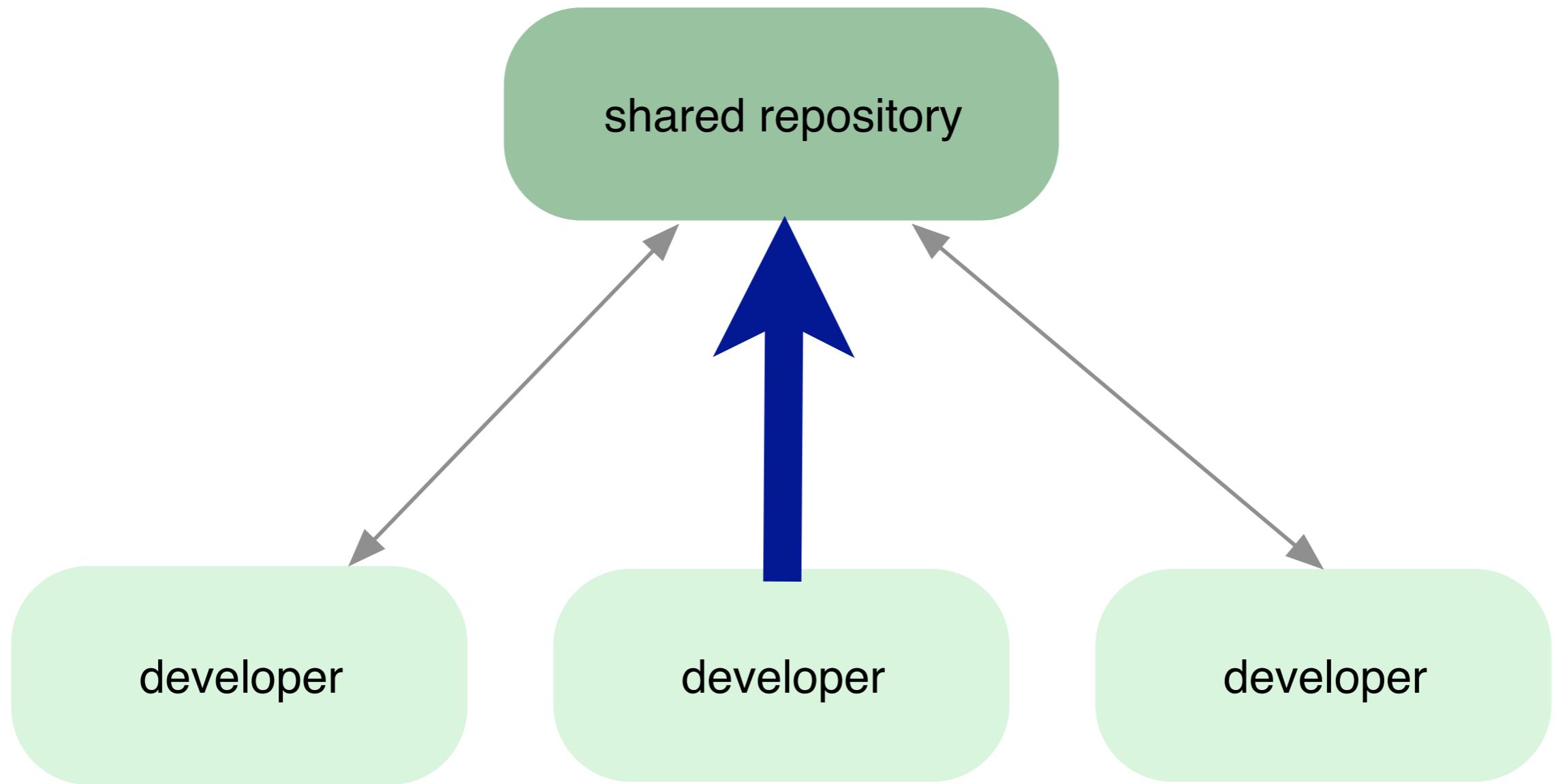


git push



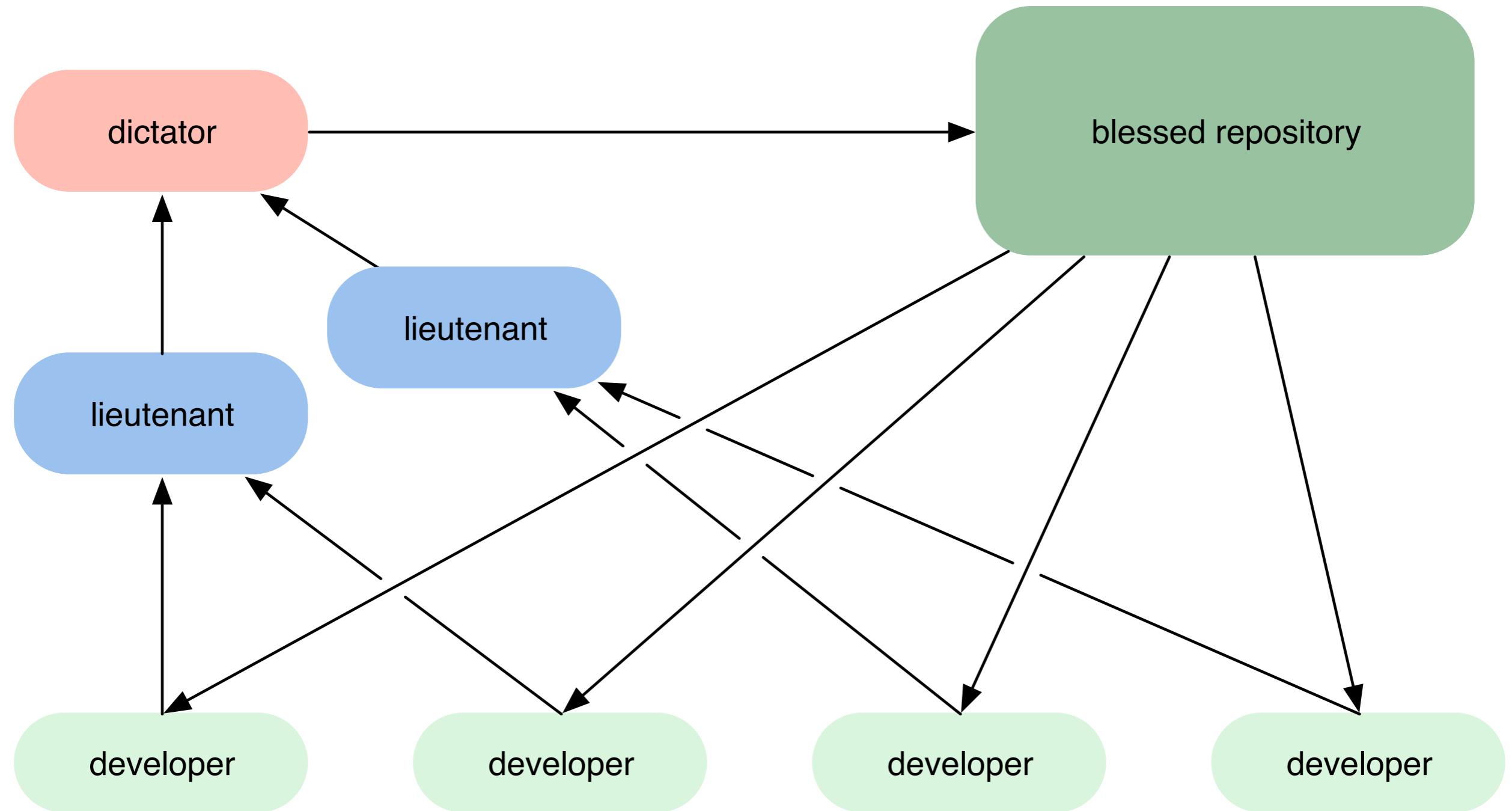


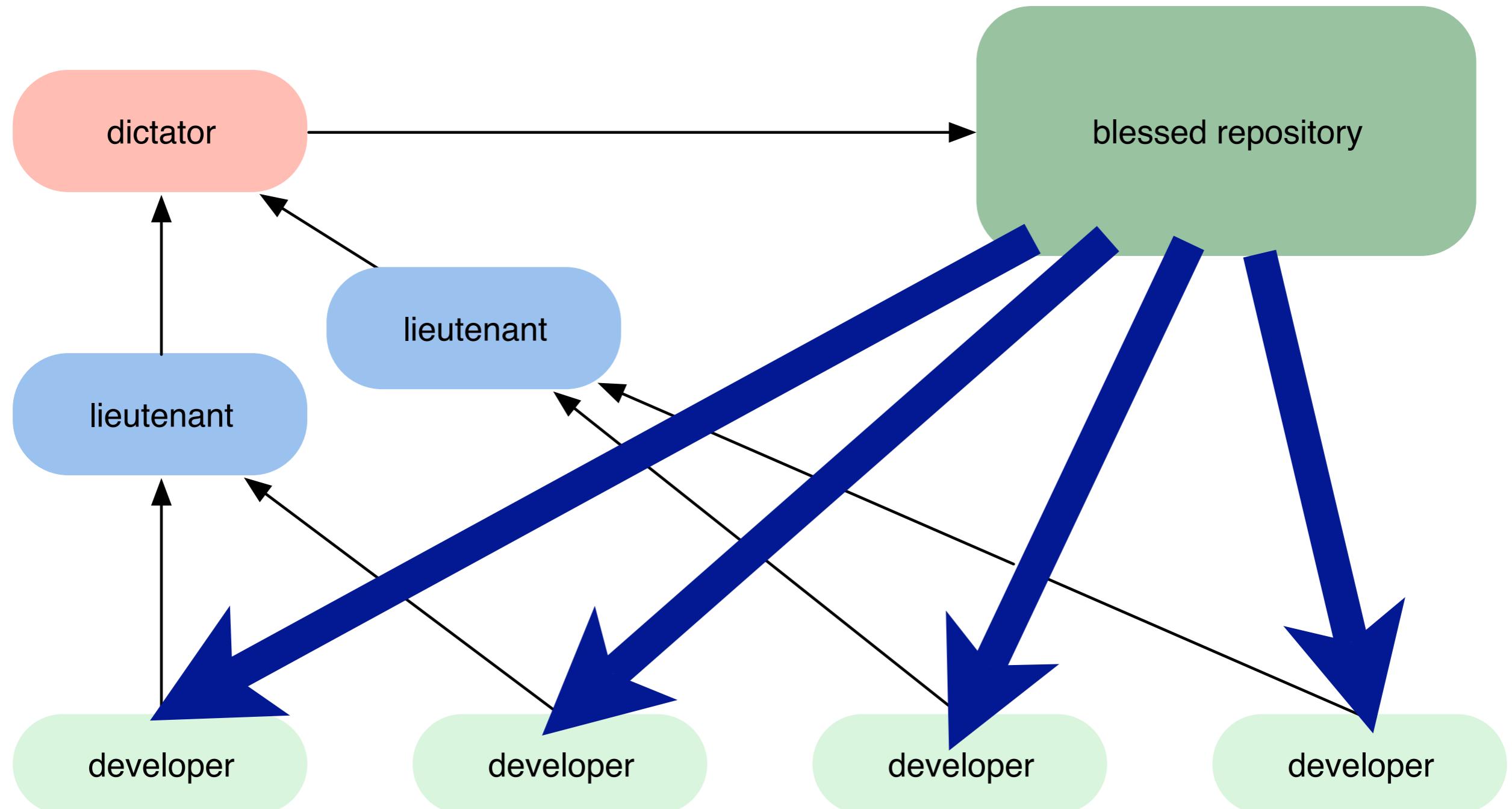
**git fetch
git merge**



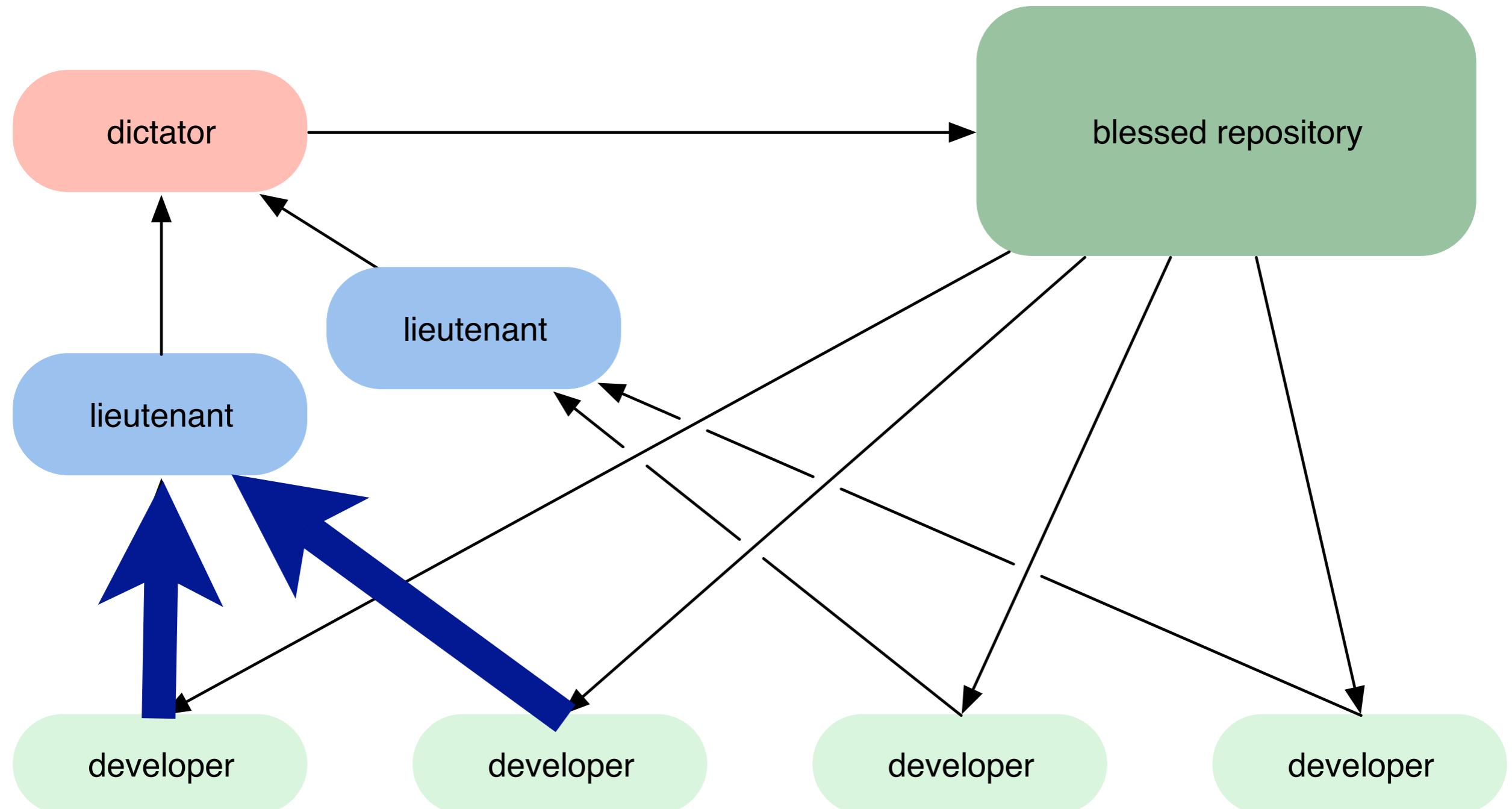
git push

**dictator and lieutenants
model**

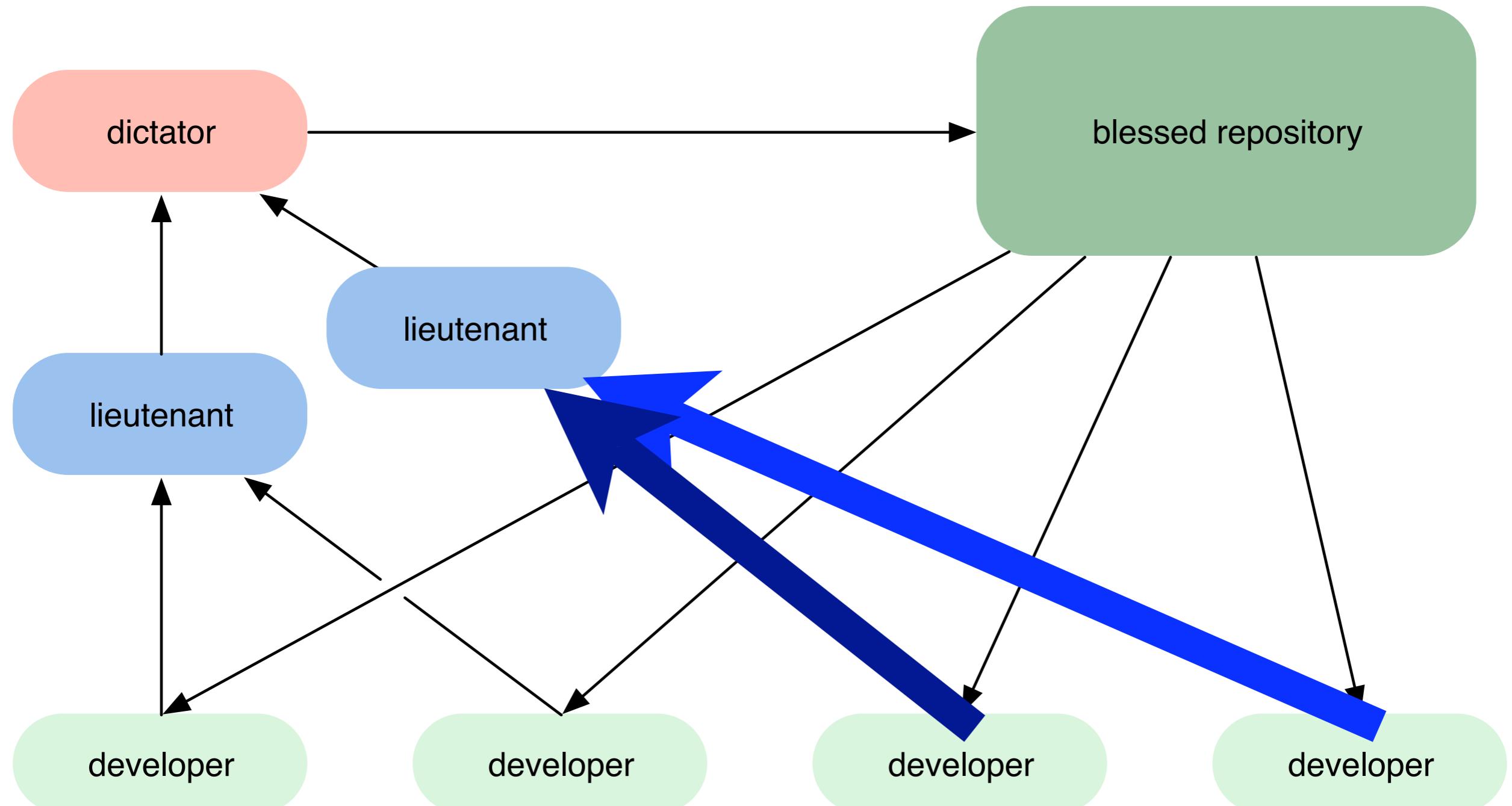




git clone

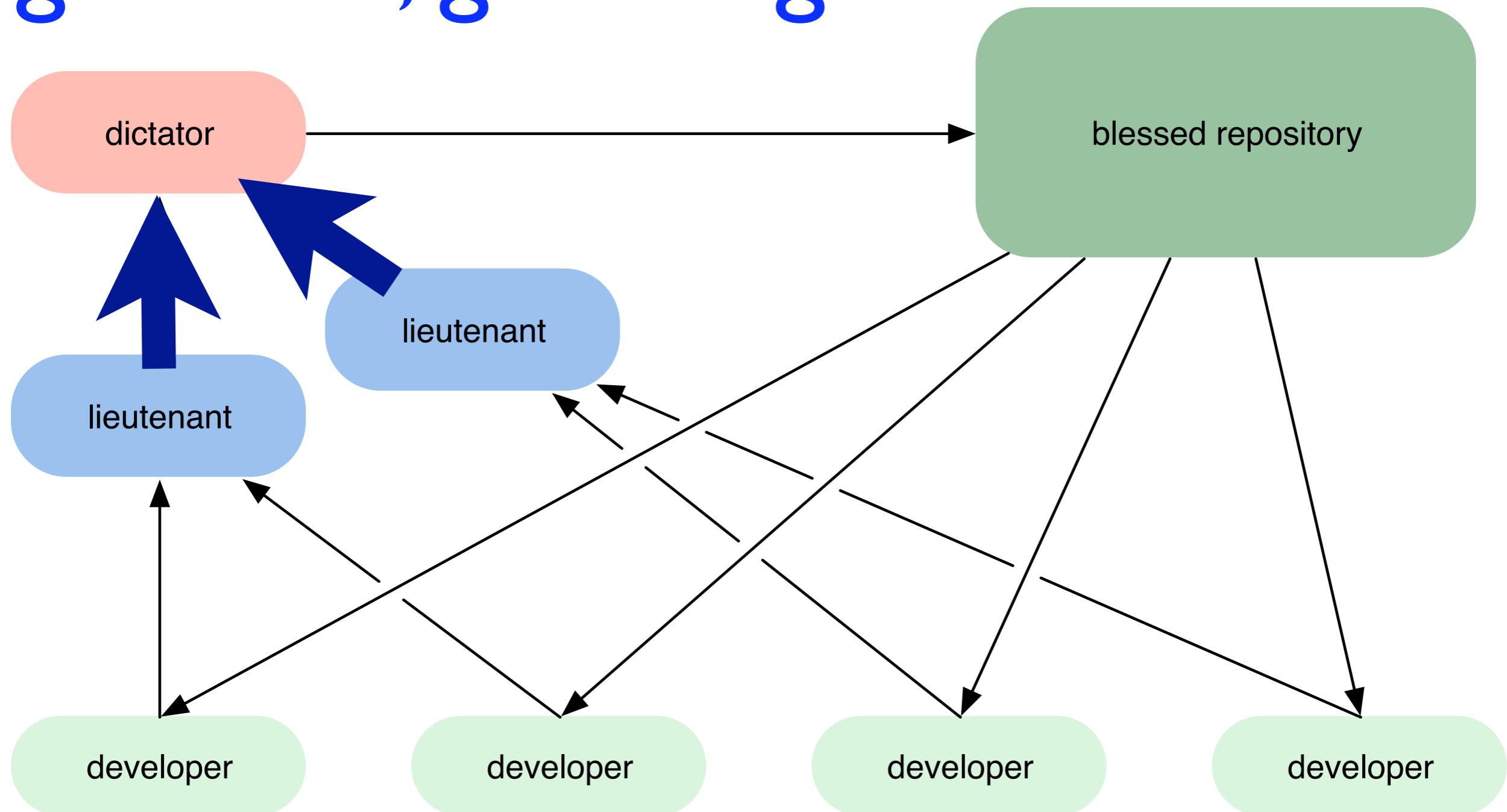


**git fetch
git merge**

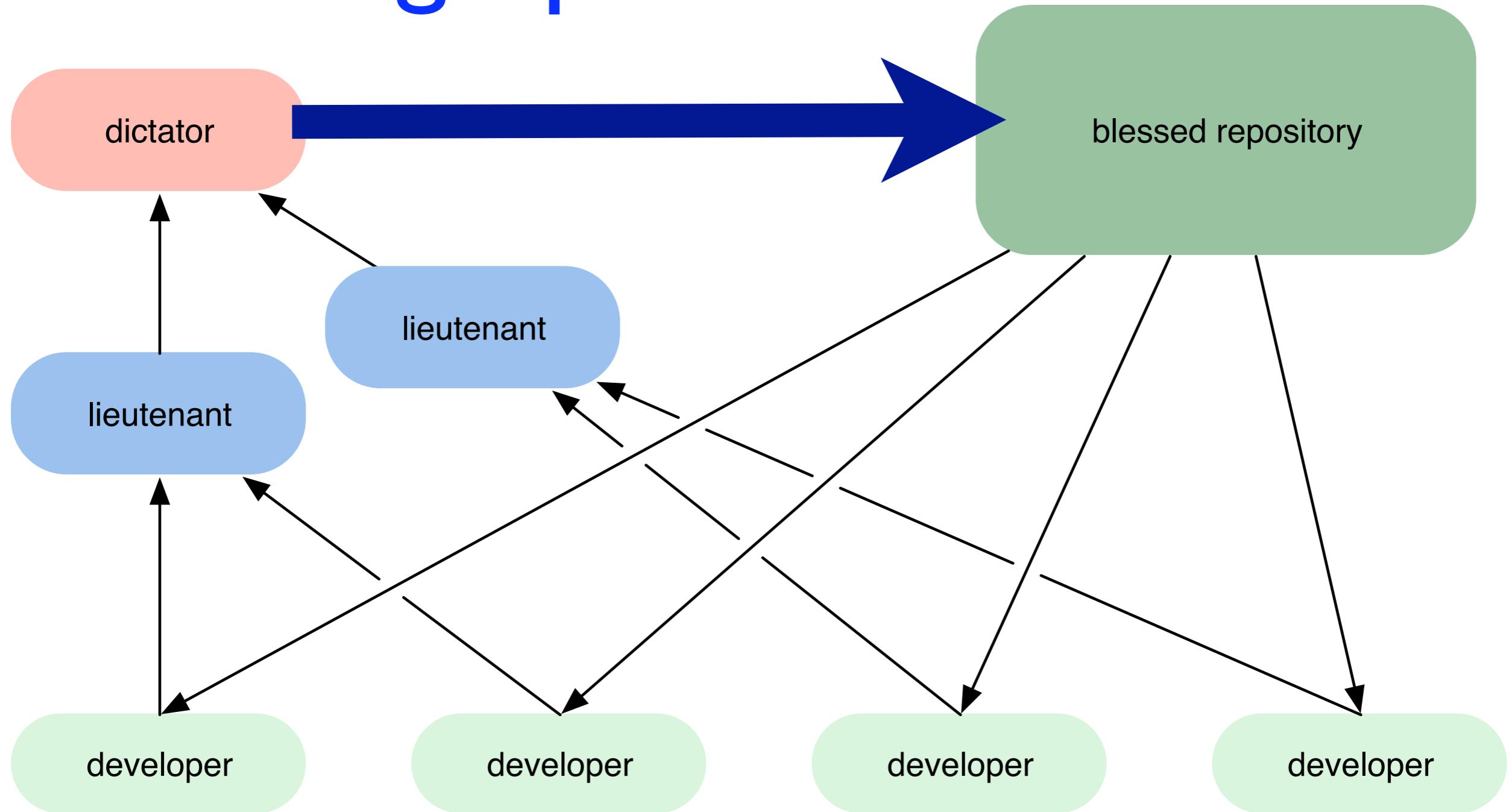


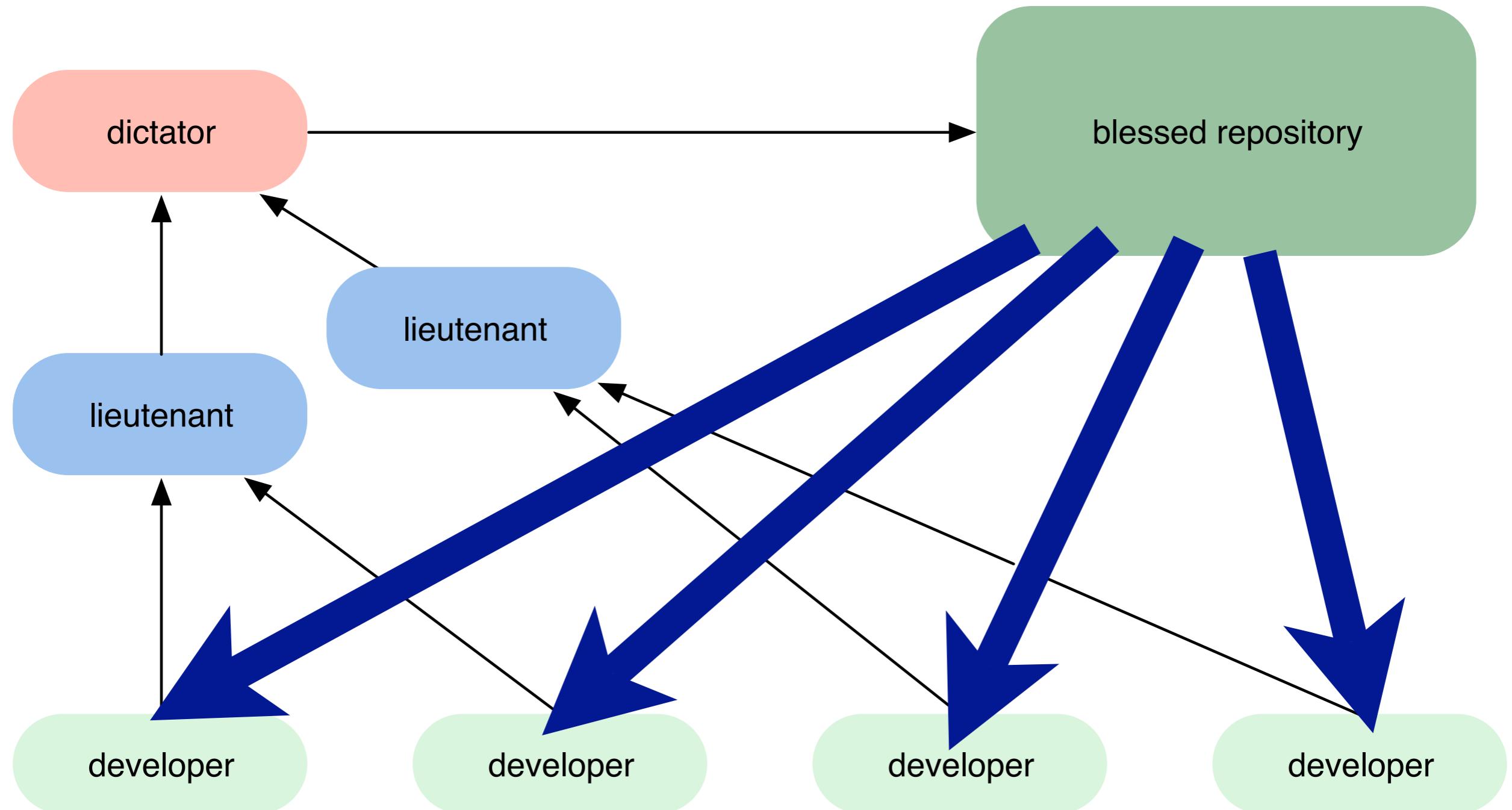
**git fetch
git merge**

git fetch; git merge



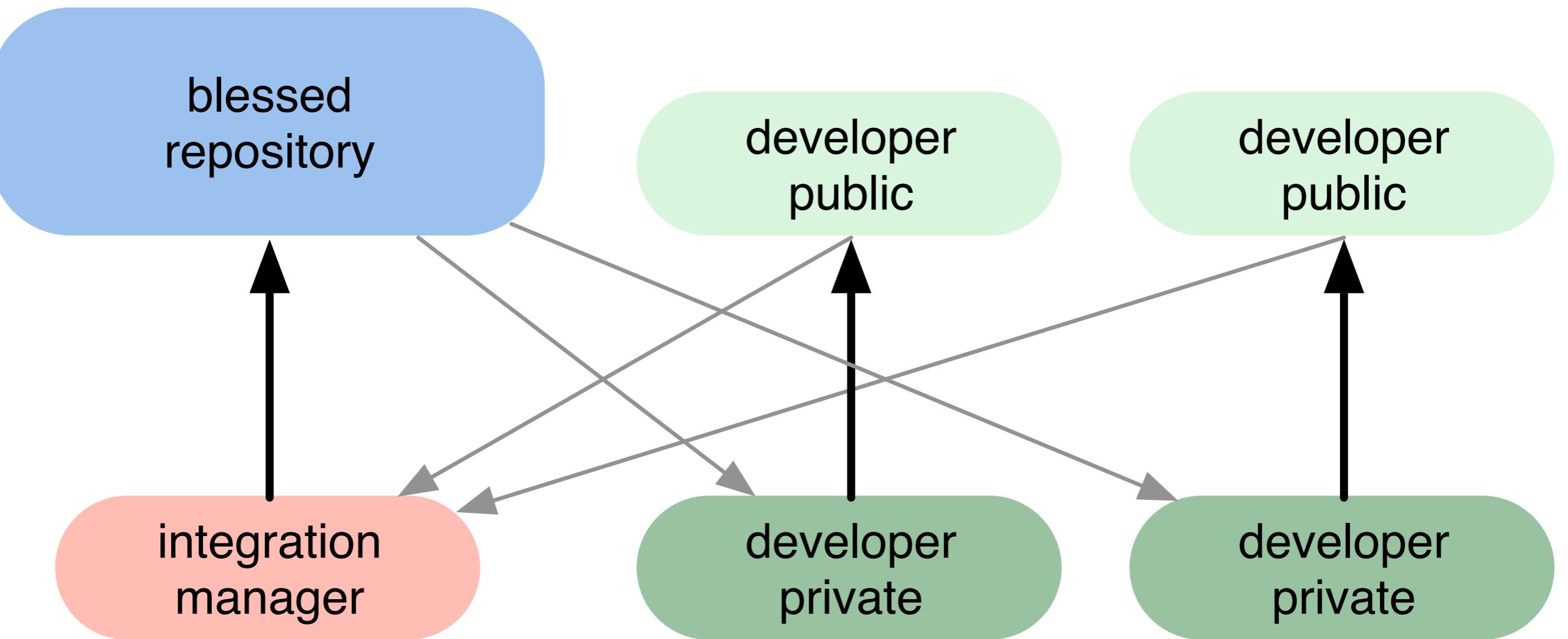
git push





git fetch

integration manager model



github.com

/schacon/ticgit

blessed
repository

/yob/ticgit

developer
public

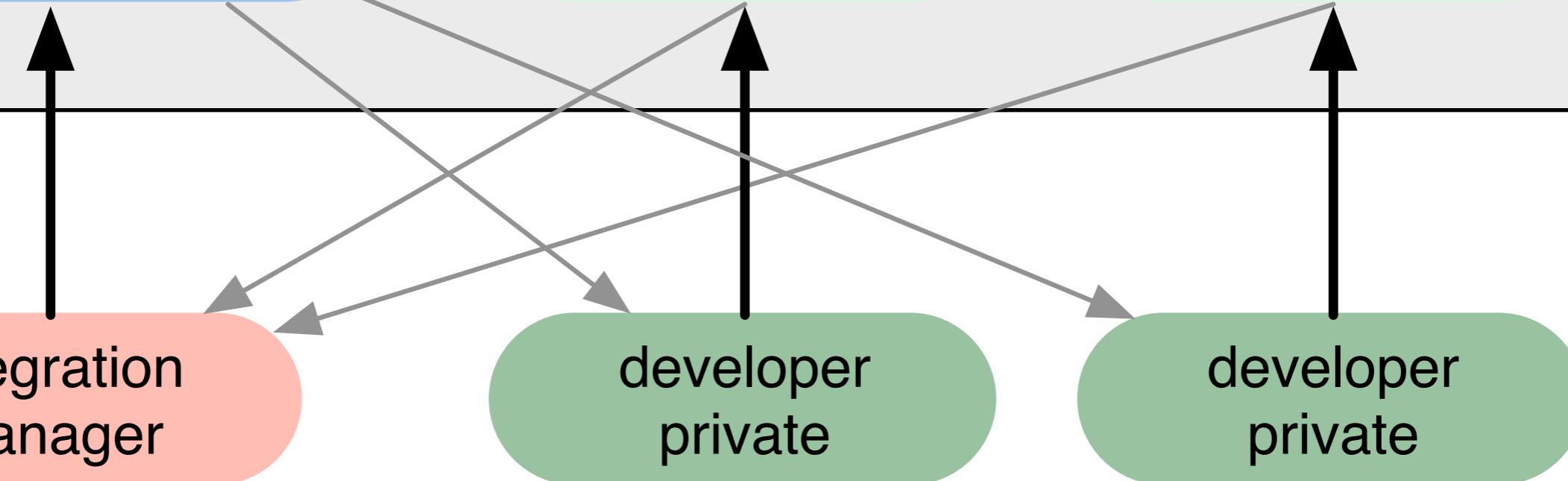
/pope/ticgit

developer
public

integration
manager

developer
private

developer
private



github.com

/schacon/ticgit

blessed
repository

“forks”

/yob/ticgit

developer
public

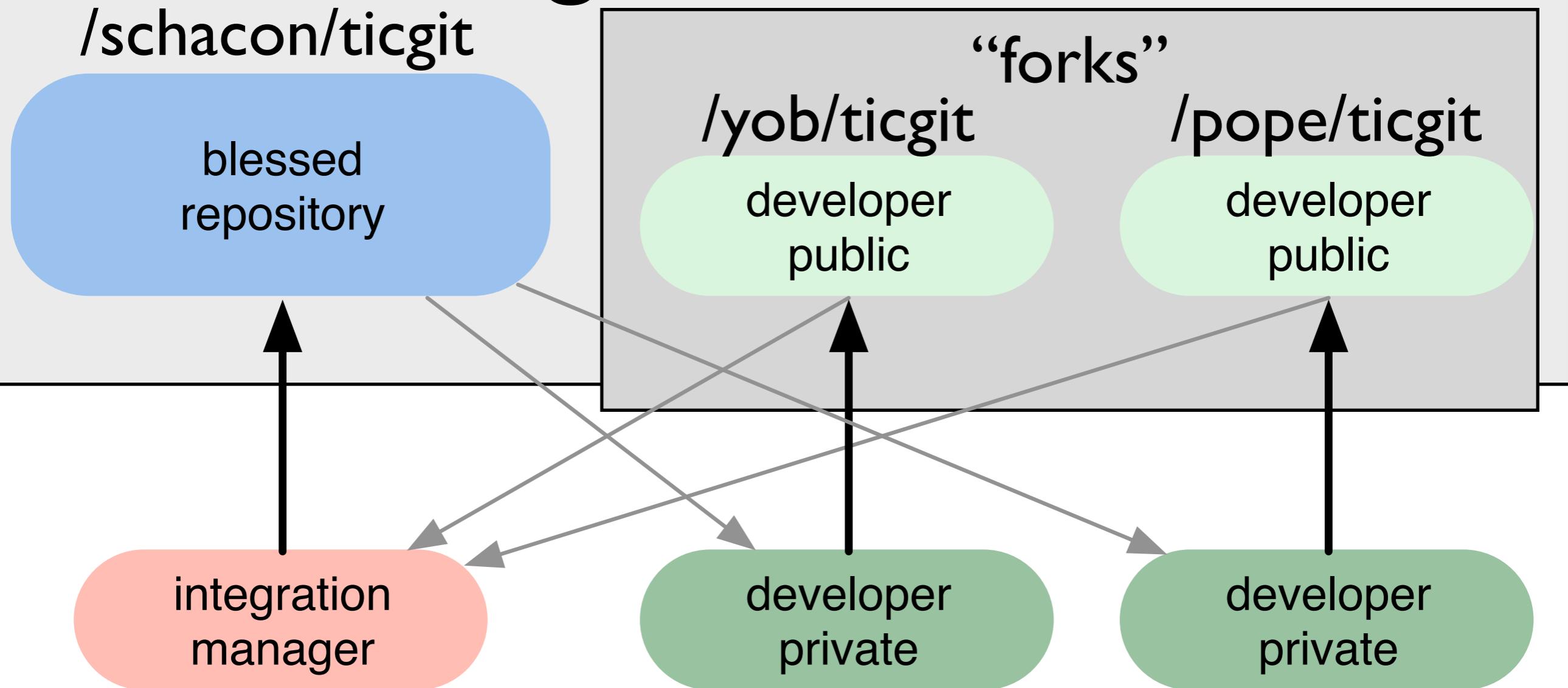
/pope/ticgit

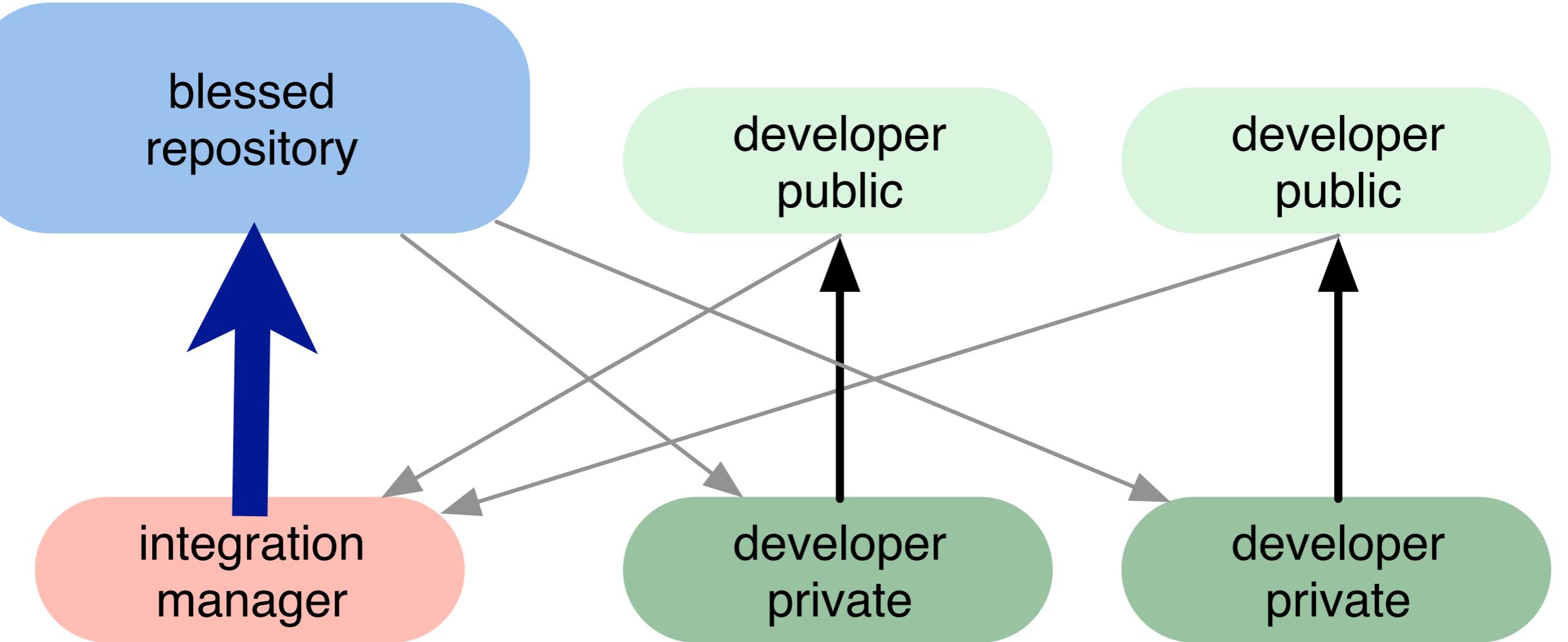
developer
public

integration
manager

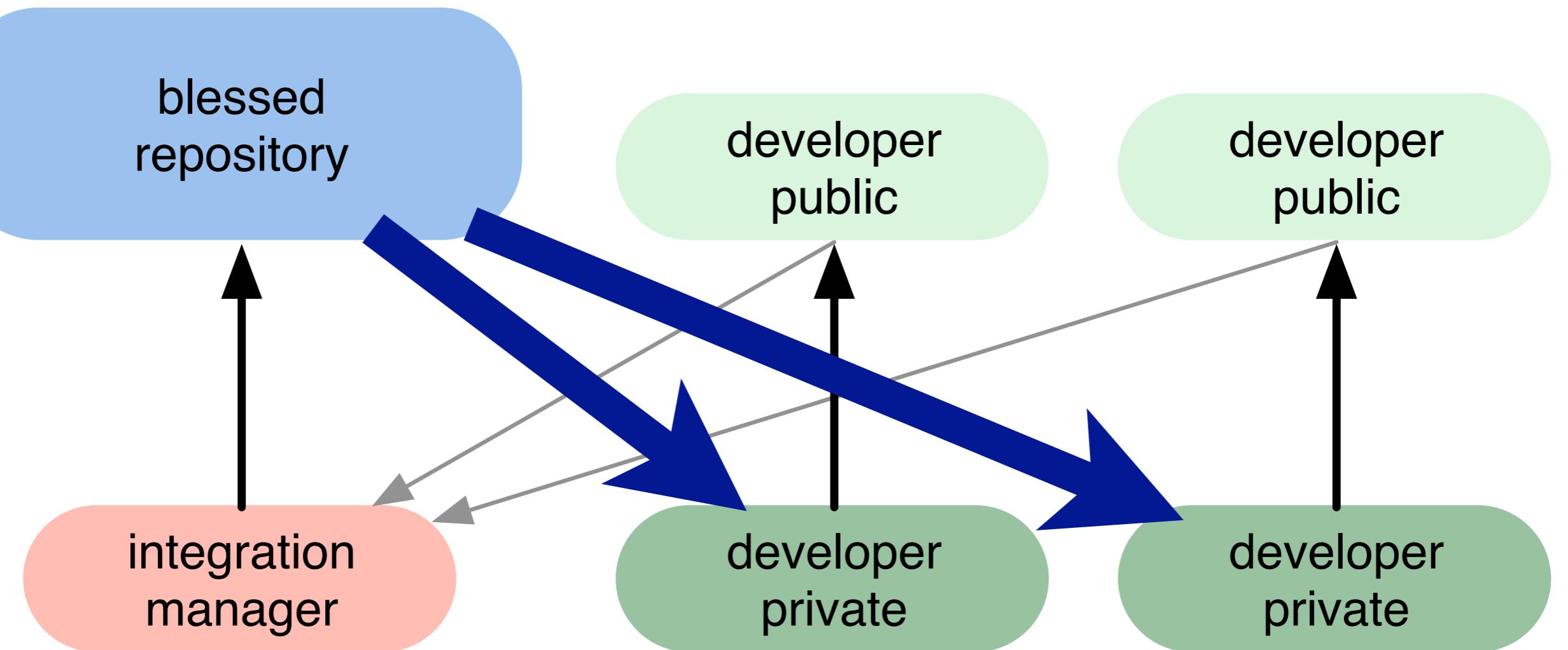
developer
private

developer
private

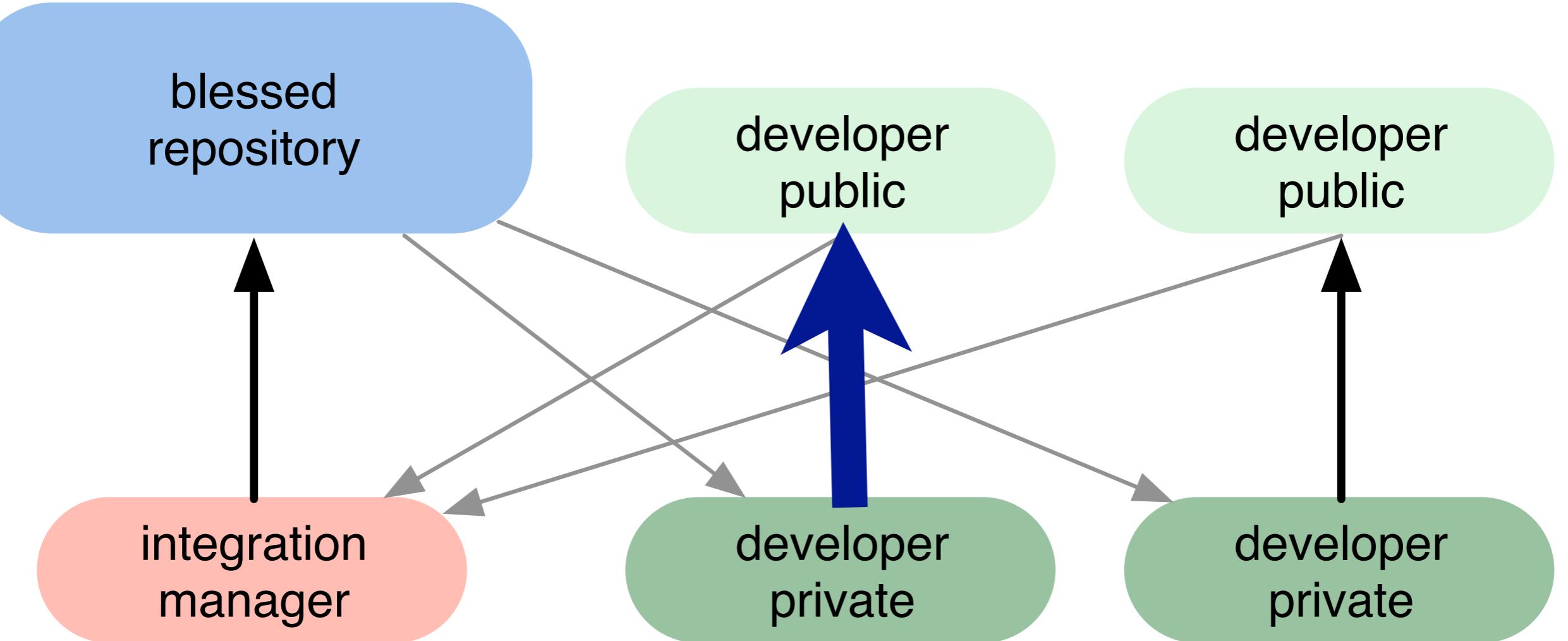




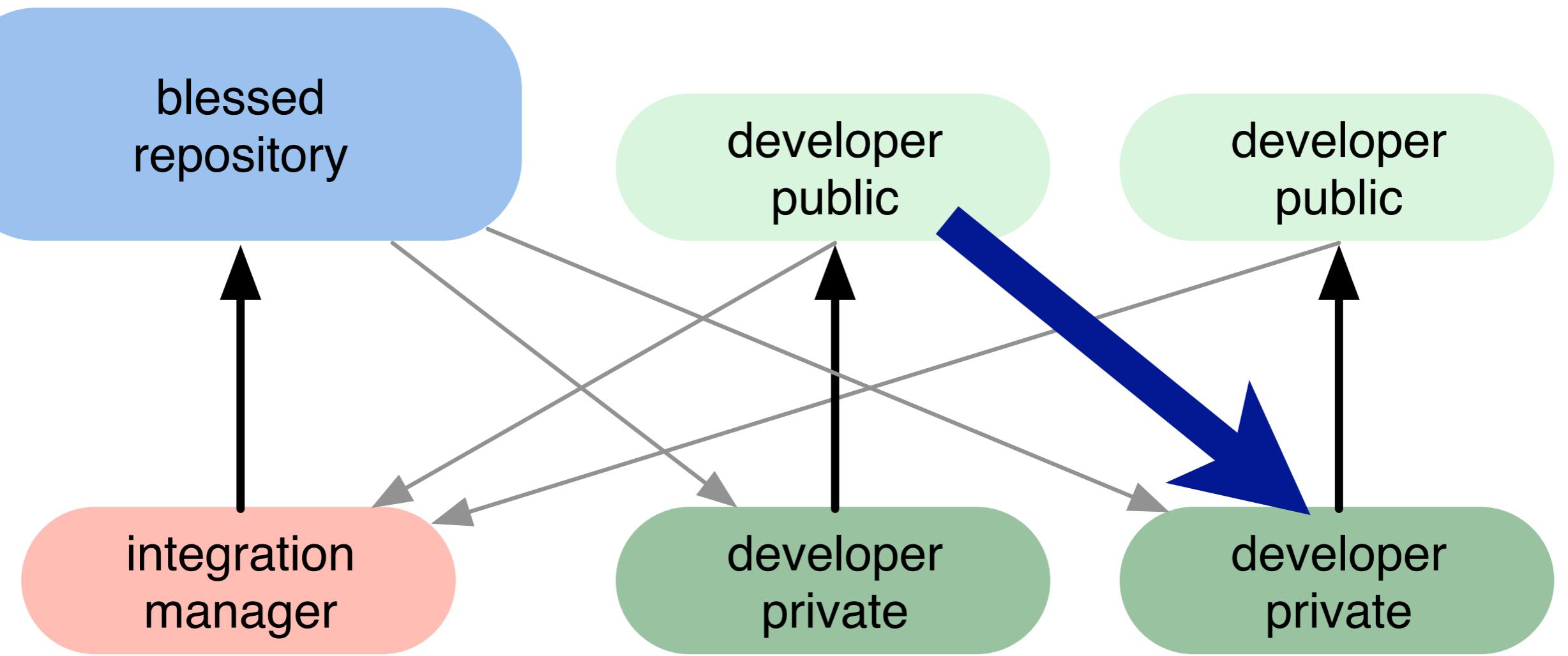
git push



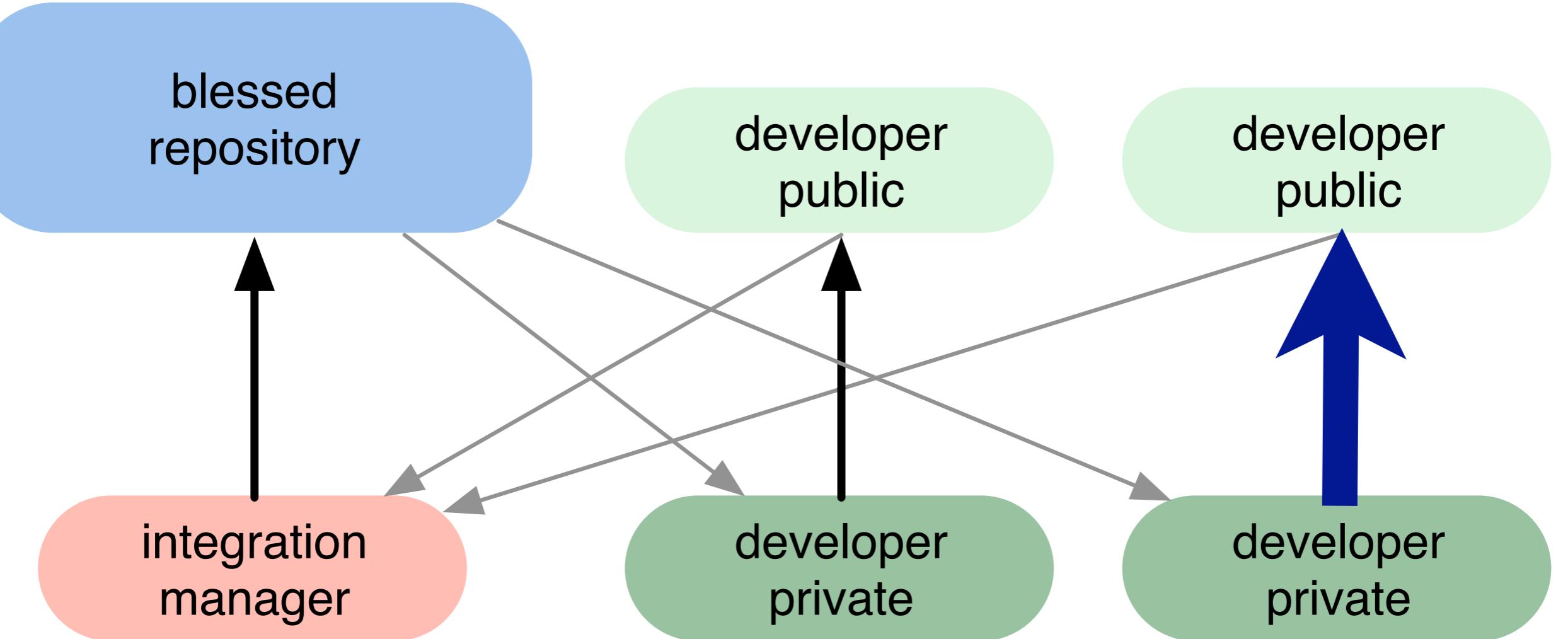
git clone



git push



**git fetch
git merge**



git push

blessed
repository

integration
manager

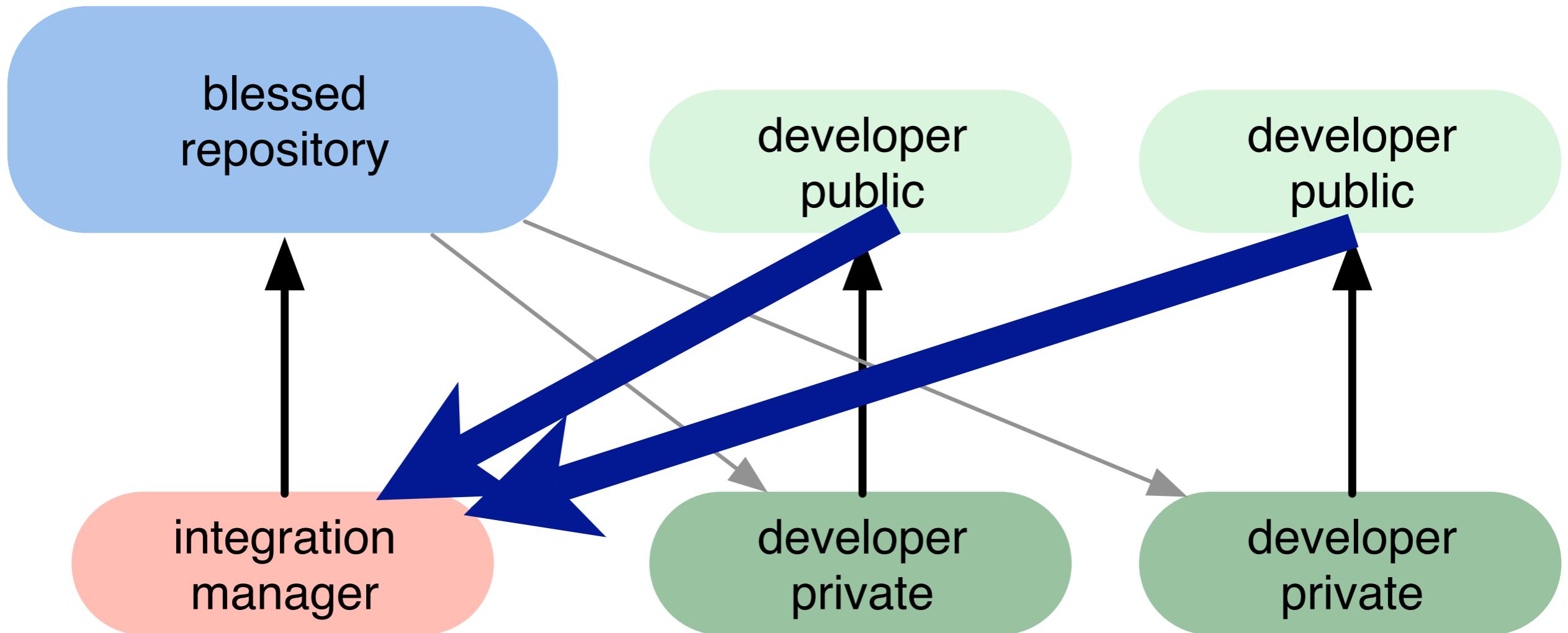
developer
public

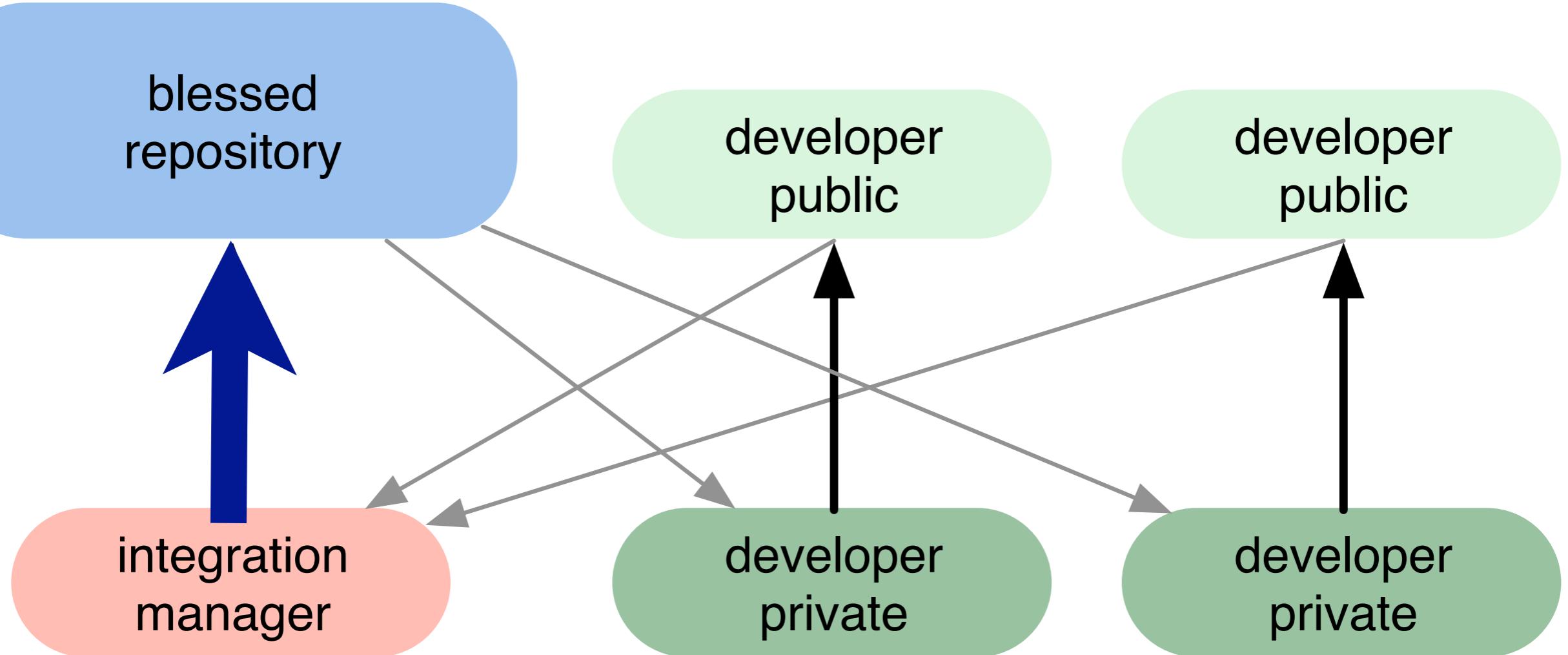
developer
private

developer
public

developer
private

git fetch
git rebase / merge





git push

How Do I Deploy with Git?

10%
4 min

2m

This Slide
Who is Scott?

75%
35 min

3m

What is Git?

20m

How Does Git Work?

12m

How Do I Use Git?

15%
6 min

4m

How Do I Deploy with Git?

2m

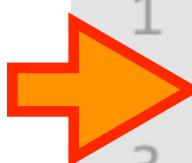
Where Can I Learn More?

capistrano + git

deploy.rb

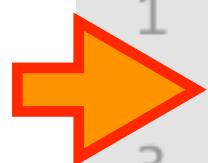
```
1 set :application, "webapp"
2 set :repository, "http://svn.ourcompany.com/myapp/trunk"
3
4 set :deploy_to, "/var/www/#{application}"
5
6 set :scm, :svn
7
8
9 set :user, 'root'
10
11 ssh_options[:paranoid] = false
12
13 set :domain, 'myserver.com'
14 role :app, domain
15 role :web, domain
16 role :db, domain, :primary => true
17
```

deploy.rb

```
1 set :application, "webapp"
 2 set :repository, "http://svn.ourcompany.com/myapp/trunk"
3
4 set :deploy_to, "/var/www/#{application}"
5
6 set :scm, :svn
7
8
9 set :user, 'root'
10
11 ssh_options[:paranoid] = false
12
13 set :domain, 'myserver.com'
14 role :app, domain
15 role :web, domain
16 role :db, domain, :primary => true
17
```

deploy.rb

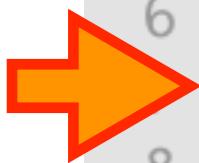
```
1 set :application, "webapp"
2 set :repository, "http://svn.ourcompany.com/myapp/trunk"
3
4 set :deploy_to, "/var/www/#{application}"
5
6 set :scm, :svn
7
8
9 set :user, 'root'
10
11 ssh_options[:paranoid] = false
12
13 set :domain, 'myserver.com'
14 role :app, domain
15 role :web, domain
16 role :db, domain, :primary => true
17
```



```
1 set :application, "webapp"
2 #set :repository, "http://svn.ourcompany.com/myapp/trunk"
3 set :repository, "ssh://git@github.com:mycompany-priv/myapp.git"
4 set :deploy_to, "/var/www/#{application}"
5
6 set :scm, :svn
7
8
9 set :user, 'root'
10
11 ssh_options[:paranoid] = false
12
13 set :domain, 'myserver.com'
14 role :app, domain
15 role :web, domain
16 role :db, domain, :primary => true
17
```



```
1 set :application, "webapp"
2 #set :repository, "http://svn.ourcompany.com/myapp/trunk"
3 set :repository, "ssh://git@github.com:mycompany-priv/myapp.git"
4 set :deploy_to, "/var/www/#{application}"
5
6 #set :scm, :svn
7 set :scm, :git
8
9 set :user, 'root'
10
11 ssh_options[:paranoid] = false
12
13 set :domain, 'myserver.com'
14 role :app, domain
15 role :web, domain
16 role :db, domain, :primary => true
17
```



```
1 set :application, "webapp"
2 #set :repository, "http://svn.ourcompany.com/myapp/trunk"
3 set :repository, "ssh://git@github.com:mycompany-priv/myapp.git"
4 set :deploy_to, "/var/www/#{application}"
5
6 #set :scm, :svn
7 set :scm, :git
8 set :deploy_via, :remote_cache
9 set :user, 'root'
10
11 ssh_options[:paranoid] = false
12
13 set :domain, 'myserver.com'
14 role :app, domain
15 role :web, domain
16 role :db, domain, :primary => true
17
```



other cap options

```
set :branch, "master"
```

```
set :git_shallow_clone, |
```

```
set :git_enable_submodules, |
```

ruby + git

ruby + git

grit

git gem

git-ruby

ruby + git

grit

git gem

git-ruby

```
1 require 'rubygems'  
2 require 'git'  
3  
4 Dir.chdir('simplegit') do  
5   g = Git.init  
6   g.add  
7   g.commit('initial import of simplegit')  
8   g.add_tag('v2.3')  
9 end  
10
```

```
1 require 'rubygems'  
2 require 'git'  
3  
4 g = Git.clone('git://github.com/schacon/simplegit.git', 'sg')  
5 g.config('user.name', 'Scott Chacon')  
6 g.config('user.email', 'email@email.com')  
7  
8 g.log(10).each do |commit|  
9   commit.parent.sha  
10  commit.parents.size  
11  commit.author.name  
12  commit.author.email  
13  commit.author.date.strftime("%m-%d-%y")  
14  commit.committer.name  
15  commit.date.strftime("%m-%d-%y")  
16  commit.message  
17  
18  tree = commit.gtree  
19  tree.blobs  
20  tree.subtrees  
21  tree.children # blobs and subtrees  
22 end  
23
```



**github.com/schacon/
ruby-git**

**Where Can I
Learn More?**

10%
4 min

2m

This Slide
Who is Scott?

75%
35 min

3m

What is Git?

20m

How Does Git Work?

12m

How Do I Use Git?

15%
6 min

4m

How Do I Deploy with Git?

2m

Where Can I Learn More?

Resources

git.or.cz

gitcasts.com

del.icio.us/popular/git

#git / #github on IRC

peepcode - git book and screencast

schacon@gmail.com

me. here.

thats it!

thats it!

gitcasts.com/git-talk

notes, these slides, etc

thats it!

gitcasts.com/git-talk

notes, these slides, etc

git.or.cz

del.icio.us/popular/git

peepcode

gitcasts.com

#git / #github on IRC

schacon@gmail.com