

Git going with DVCS

60 minutes of
source code control magic



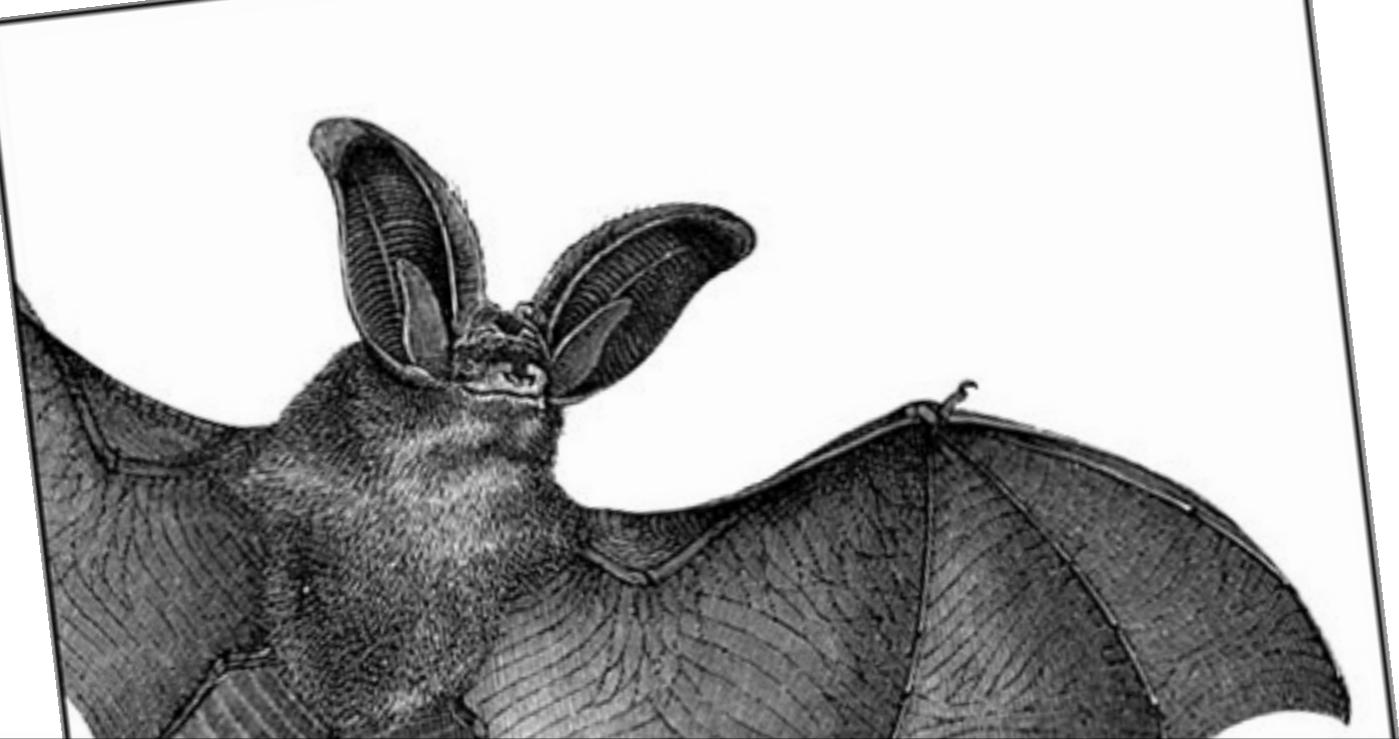
Hello
my name is

Matthew
McCullough



github
SOCIAL CODING

<http://github.com/training/online>

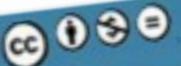


<http://orei.ly/ogitvid>

Mastering Git

Matthew McCullough
& Tim Berglund

VIDEO



Wakaleo Consulting

Optimizing your software development process

Automate your software development process

Continuous Integration with Hudson



<http://bit.ly/jenkinsbook>

John Ferguson Smart

delicious

Home Bookmarks People Tags

Search Delicious

Matthew McCullough's git Bookmarks

Bookmarks | Network | Tags | Subscriptions | Inbox

See all git bookmarks in Popular, Recent, or your Network.

matthew.mccullough > git Type another tag

Bookmarks 118

Display options

06 APR 10 Repopular | Code Worth Mentioning - GitHub Repos 34

opensource programming git

EDIT | DELETE

02 APR 10 brotherbard's gitx experiment 9

Experimental fork of Pieter's nice git GUI for OS X. Includes: sidebar, fetch, pull, push, add remote, merge, cherry-pick, rebase, clone, clone to.

macosx opensource freeware sourcecodecontrol

programming gui git

EDIT | DELETE

29 MAR 10 Git Client - SmartGit 246

sourcecodecontrol gui git

EDIT | DELETE

tig, the ncurses front-end to Git 158

Test based GUI for Git

sourcecodecontrol gui git

EDIT | DELETE

http://delicious.com/matthew.mccullough/git

Tags People Options

Related Tags 77

- +agile 2
- +animation 1
- +apache 1
- +bash 5
- +blog 3
- +bugtracking 10
- +cheatsheet 1
- +clearcase 1
- +cocoa 1
- +collaboration 1
- +comics 2
- +continuousintegration 2
- +converter 1
- +development 6
- +documentation 1

Bookmarks 651

+free

+freeware

sourcecodecontrol howto tutorial programming git

EDIT | DELETE

28 MAR 10 A tour of git: the basics

sourcecodecontrol howto tutorial programming git

EDIT | DELETE

[Add links to this bundle](#) Click here to rearrange or delete links in your bundle[Add to bundle](#)

Matthew's Git Links

Want to give everyone an overview of what this bundle is all about? Click here to add a description of what's inside.

0 clicks

[Delete all tags](#)

```
for t in `git tag`  
do  
  git push origin :$t  
  git tag -d $t  
done
```

[git-deletealltags.bsh](#) [view raw](#)

This Gist brought to you by GitHub.

■ <https://gist.github.com/898798>

<http://bit.ly/gGv3sz> - [Info Page+](#)

Want to make this link more interesting? Click here to add your own description

339

McCullough and Berglund on Mastering Git - O'Reilly Media

<http://bit.ly/gitlinks>

■ <http://oreilly.com/catalog/0636920017462/>

<http://oreil.ly/ghgCKq> - [Info Page+](#)

Bundle stats - [Preview](#)



[matthewmccullough](#) - Owner

Views: 127

Created: 1 month ago

Last Modified: 2 hours ago

Short link: bit.ly/eyBhHo [copy](#)

Privacy Setting: [Public](#) | [Private](#)

Facebook:

[Collaborate](#)[Share](#)

Sources

[github.com](#) (3)

[progit.org](#) (1)

[www.youtube.com](#) (2)

[www.delicious.com](#) (1)

[www.nofluffjuststuff.com](#) (1)

[Show All 8](#)

Bundles by matthewmccullough

Created: 02/17/2011 3 links

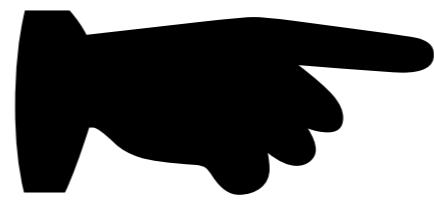
[Matthew's Git Links](#) 13 links

quit

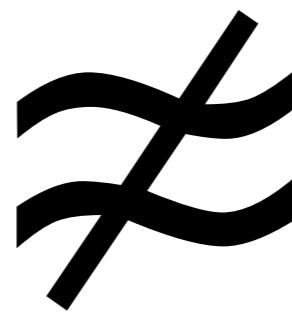
"cool kids" version control system

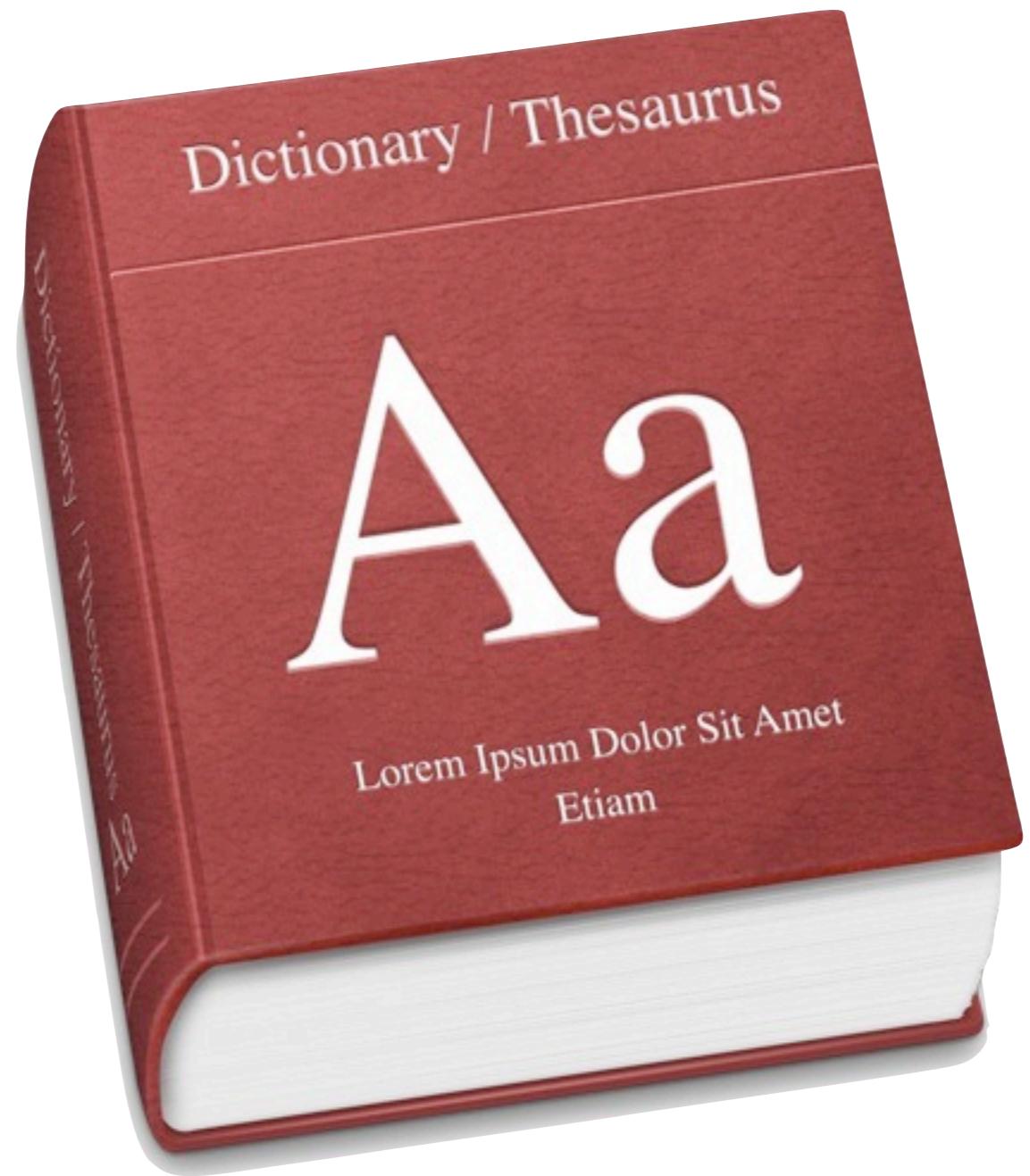
Open Source

bash scripts



C code





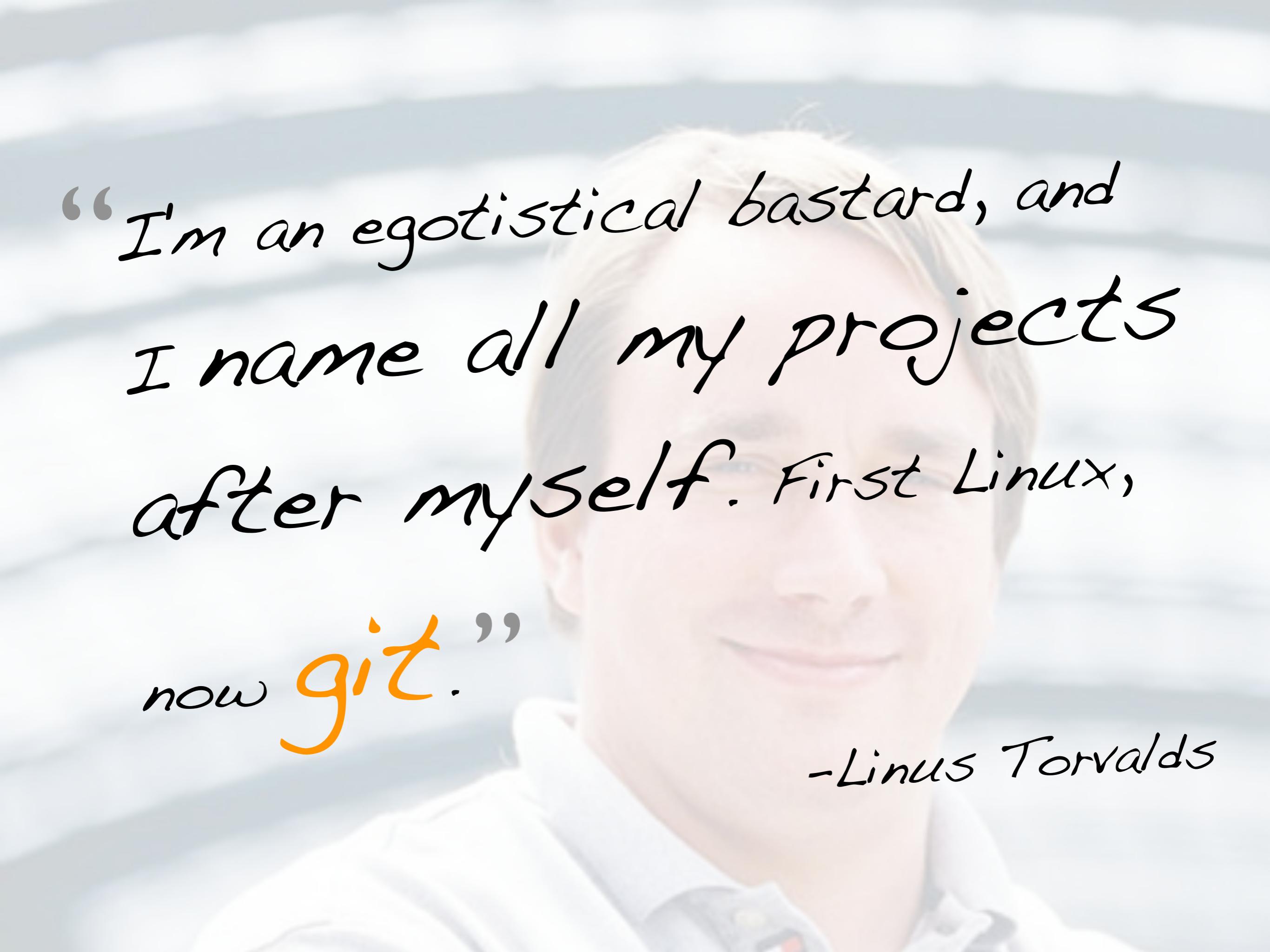
“Git

-noun

British Slang. an **unpleasant**
or contemptible person”

-Oxford English Dictionary



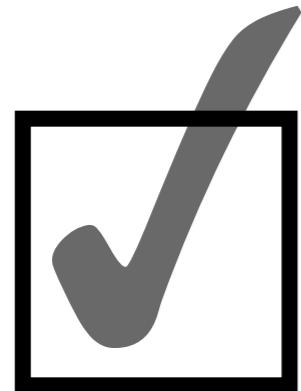
A portrait of Linus Torvalds, a man with short brown hair and a beard, wearing a light-colored shirt. He is looking slightly to his left with a neutral expression.

“I'm an egotistical bastard, and
I name all my projects
after myself. First Linux,
now git.”

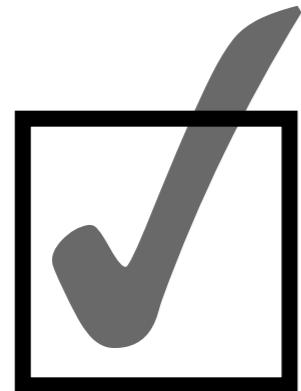
-Linus Torvalds



Coming from Subversion?



Coming from Subversion?



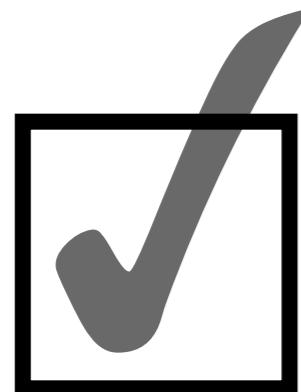
Coming from Subversion?



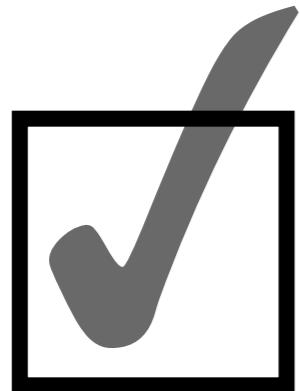
Tried any DVCS?



Coming from Subversion?



Tried any DVCS?



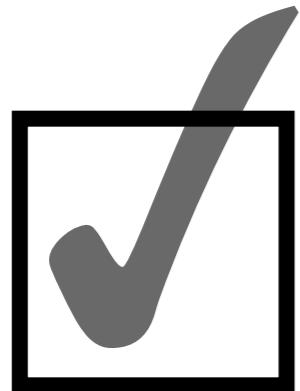
Coming from Subversion?



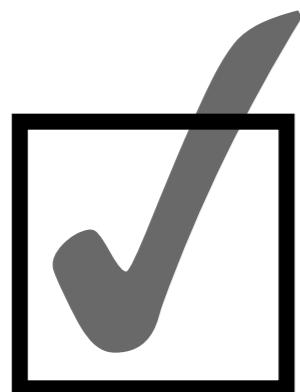
Tried any DVCS?



Using Git today?



Coming from Subversion?



Tried any DVCS?



Using Git today?



A large, yellow sunburst graphic with a white outline, featuring a jagged, serrated edge. It is positioned on the left side of the image, containing the text.

**Certified
Git User**

| 45+ commands

I 8 used on
a daily basis

init

checkout

bisect

log

revert

clone

rebase

pack

diff

merge

pull

grep

commit

push

tag

add

status

branch

Distributed Version Control

A new idea?

1997

code co-op

2001

arch

2003

monotone

2003

SVK

2003

darcs

2005

bazaar

2005

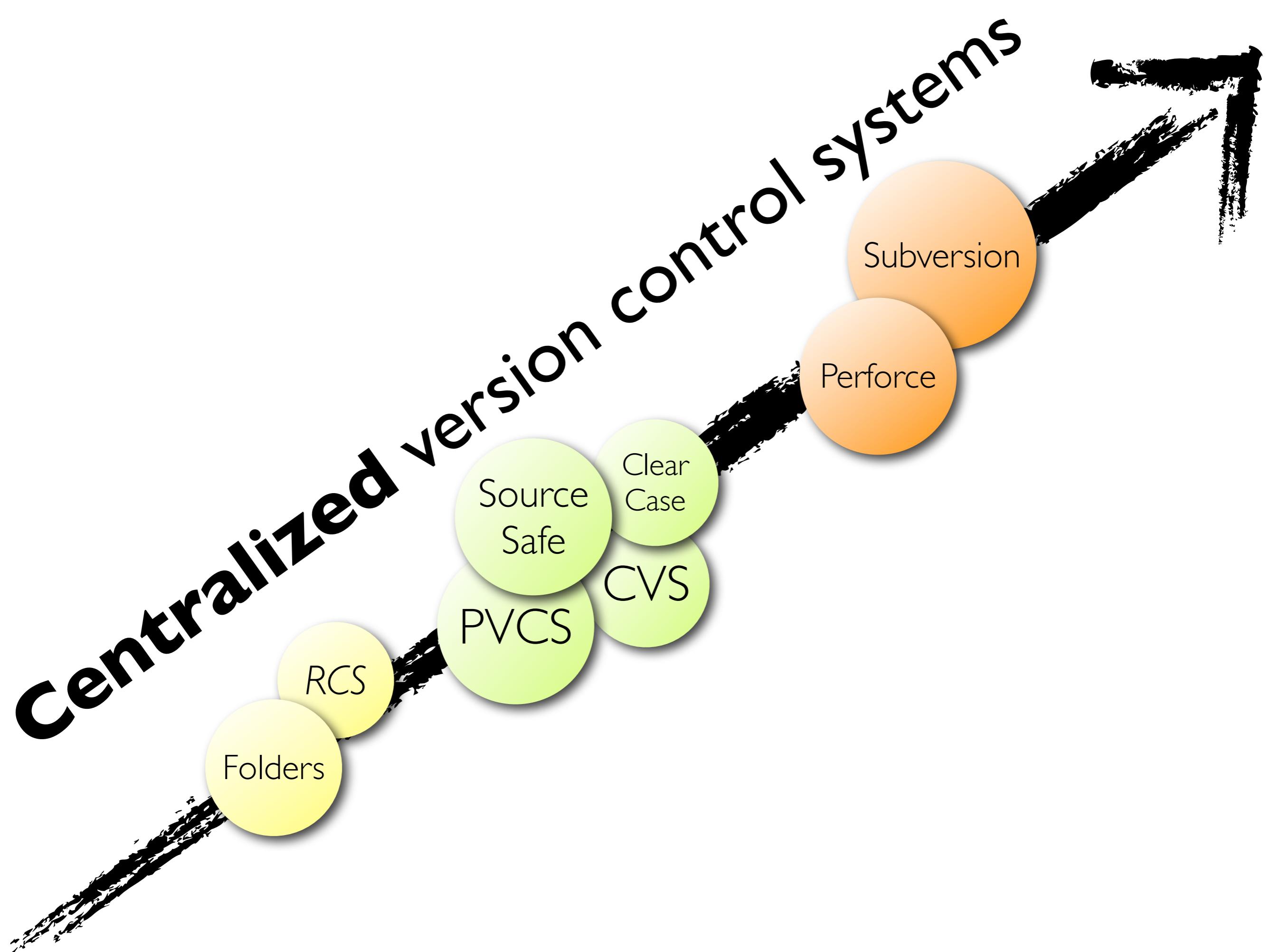
mercurial

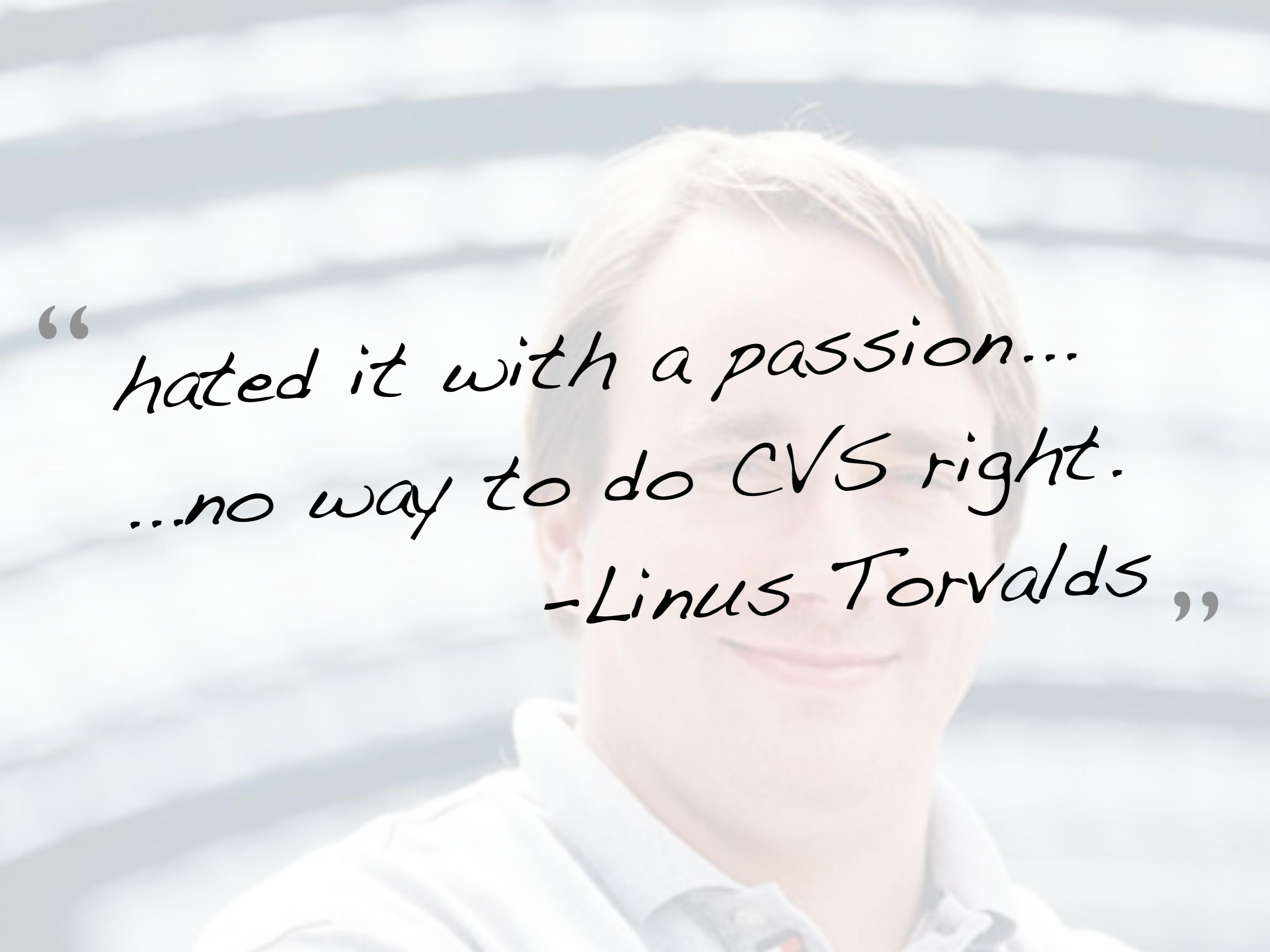
2005

git



Why?





“hated it with a passion...
...no way to do CVS right.
-Linus Torvalds ,”

HASHES AND Git



Index vs Hash

- ▶ Centralized VCS uses DB sequential index.

Index vs Hash

- ▶ Git uses SHA-1 hash.

40 hex characters

9AB223D28B1AA46EF1780B22F304982E39872C34

use as little of it as is unique

9AB223D28B1AA46EF1780B22F304982E39872C34

9AB223D28B1AA46EF1780B22F304982E39872C34

treeish

treeish = shorthand for hashes

9AB223

HEAD

HEAD[^]

HEAD~3

--since="2 weeks ago"

9AB223..56CD77

HASHABLE OBJECTS

- ▶ Blob
- ▶ Tree
- ▶ Commit
- ▶ Tag

c67db
commit

tree: a10b3
parent: nil
author: Fird
committer: Matthew
message:

Major refactoring
of the web content.

a10b3
tree

tree: 7e8b1 web
blob: 9ab16 index.html

7e8b1
tree

blob: 8d162 logo.jpg
blob: 51d22 draw.js

9ab16
blob

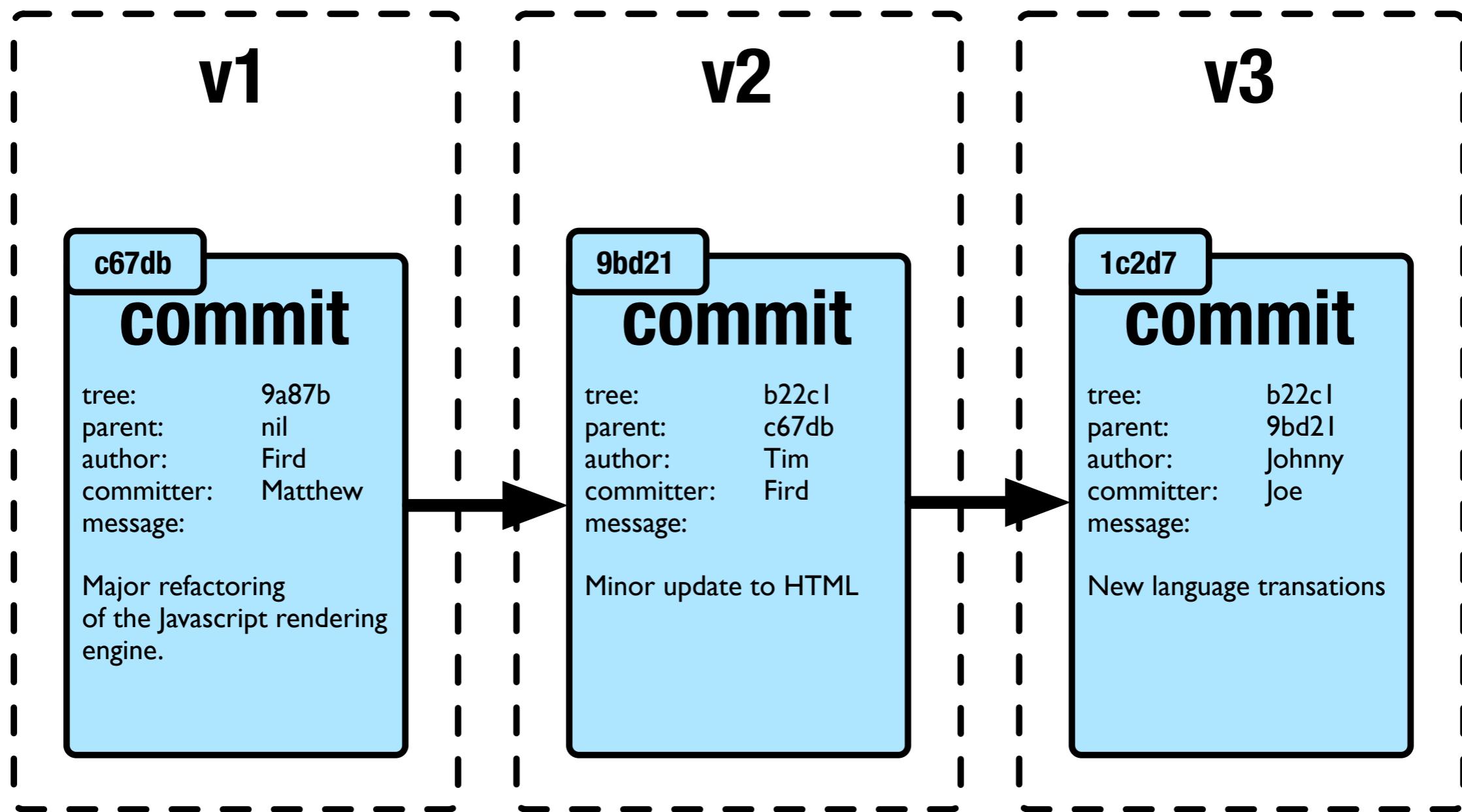
```
<html>
  <body></body>
</html>
```

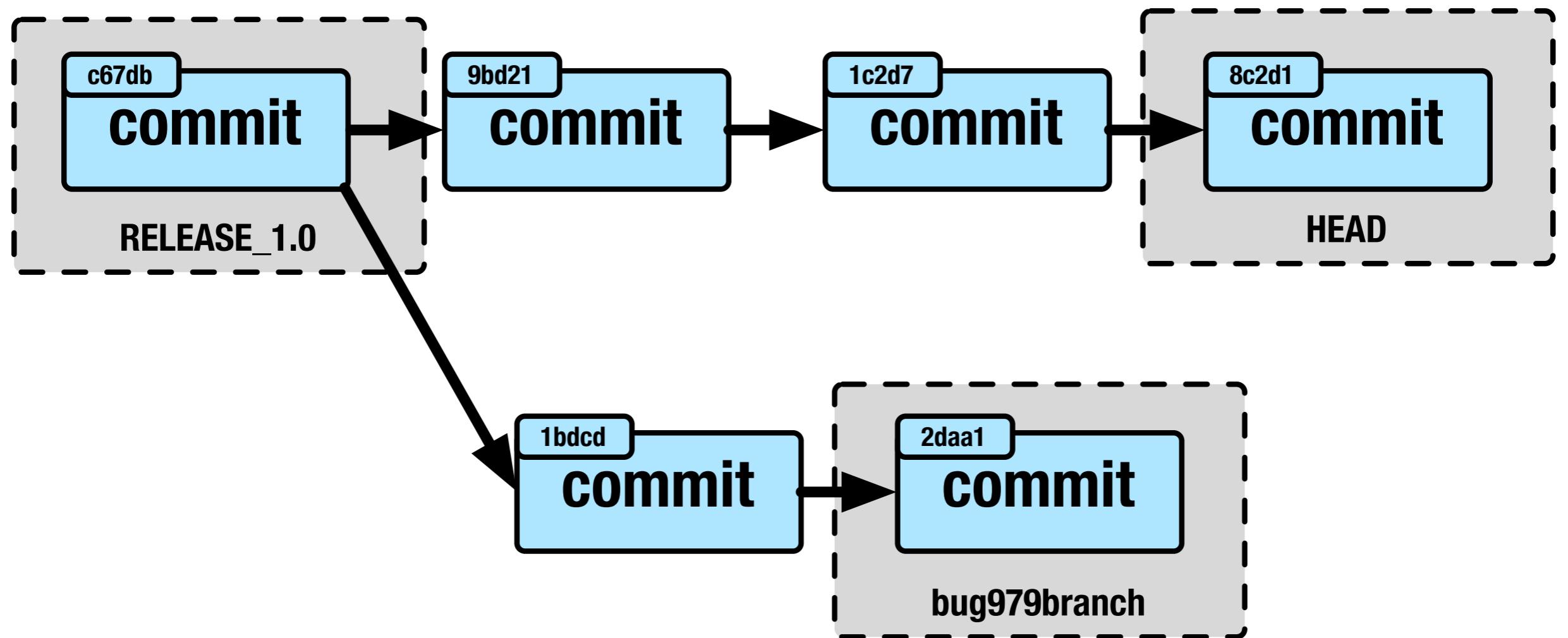
8d162
blob

7D 8D B3 7F BD 12 9F E9
7B 78 9D 3F 5C A6 72 CB

51d22
blob

```
//Some more javascript
var renderSize
```





Distributed
connectivity



developer A



developer A



developer B

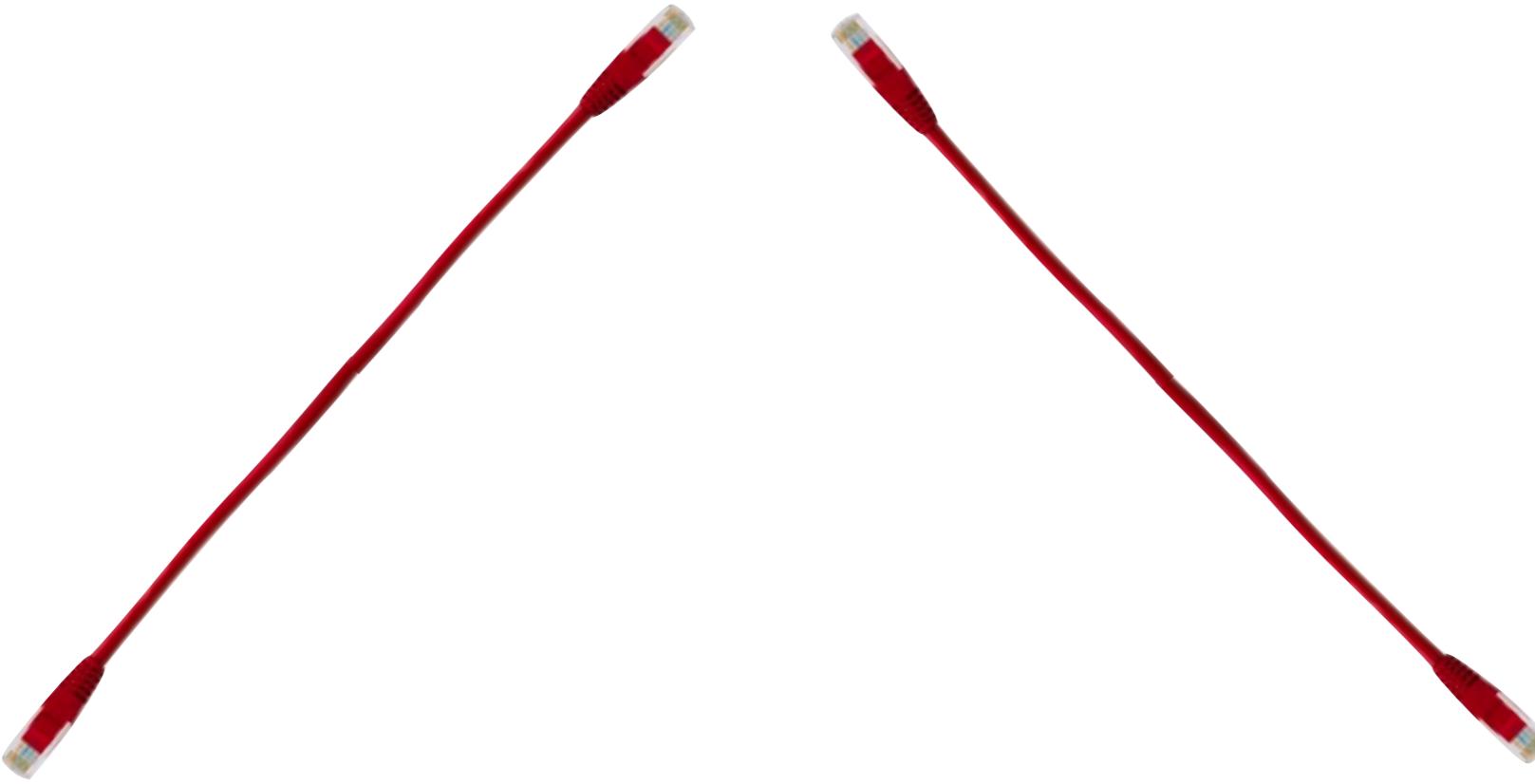


developer A



developer B

blessed repo



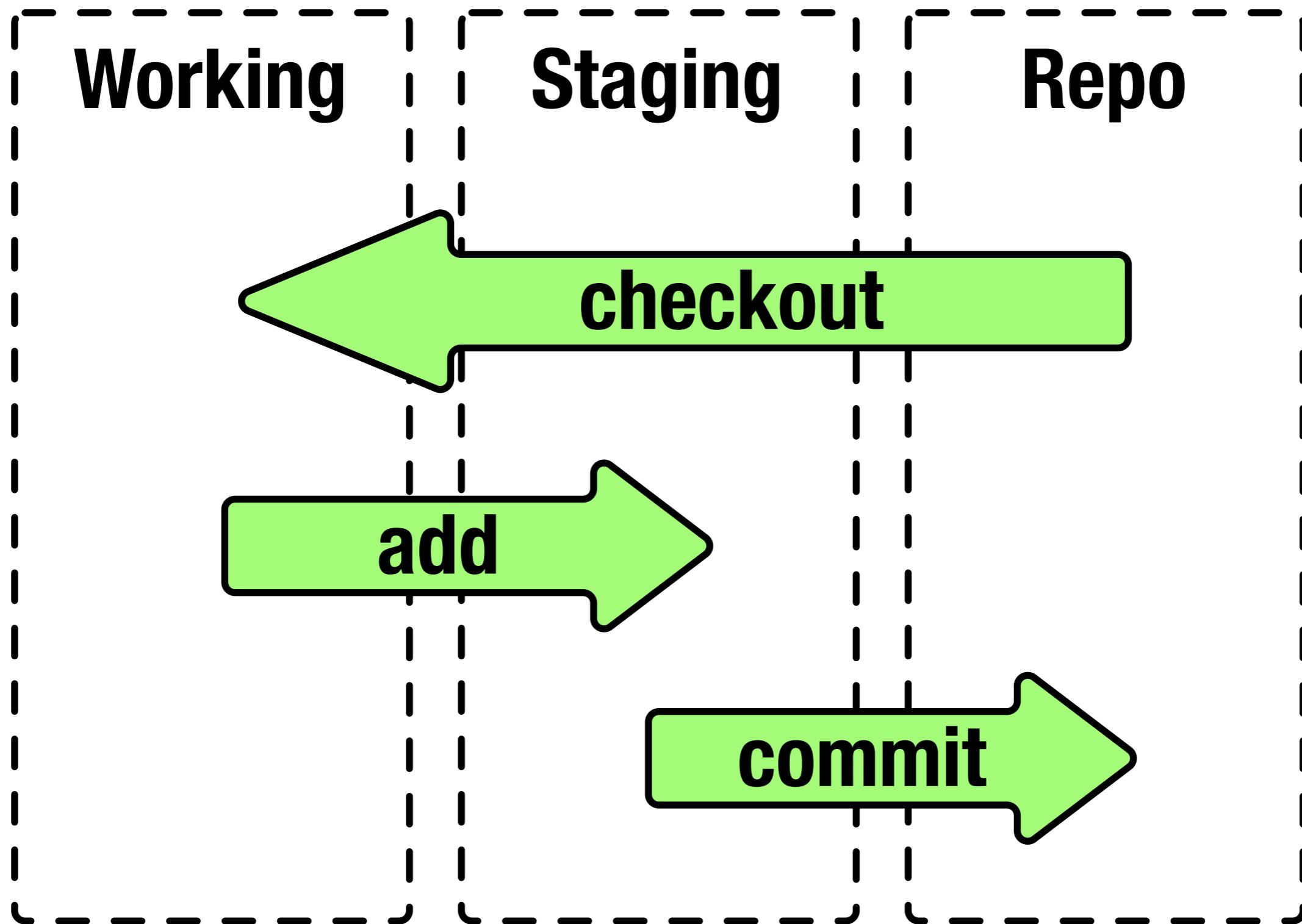
developer A

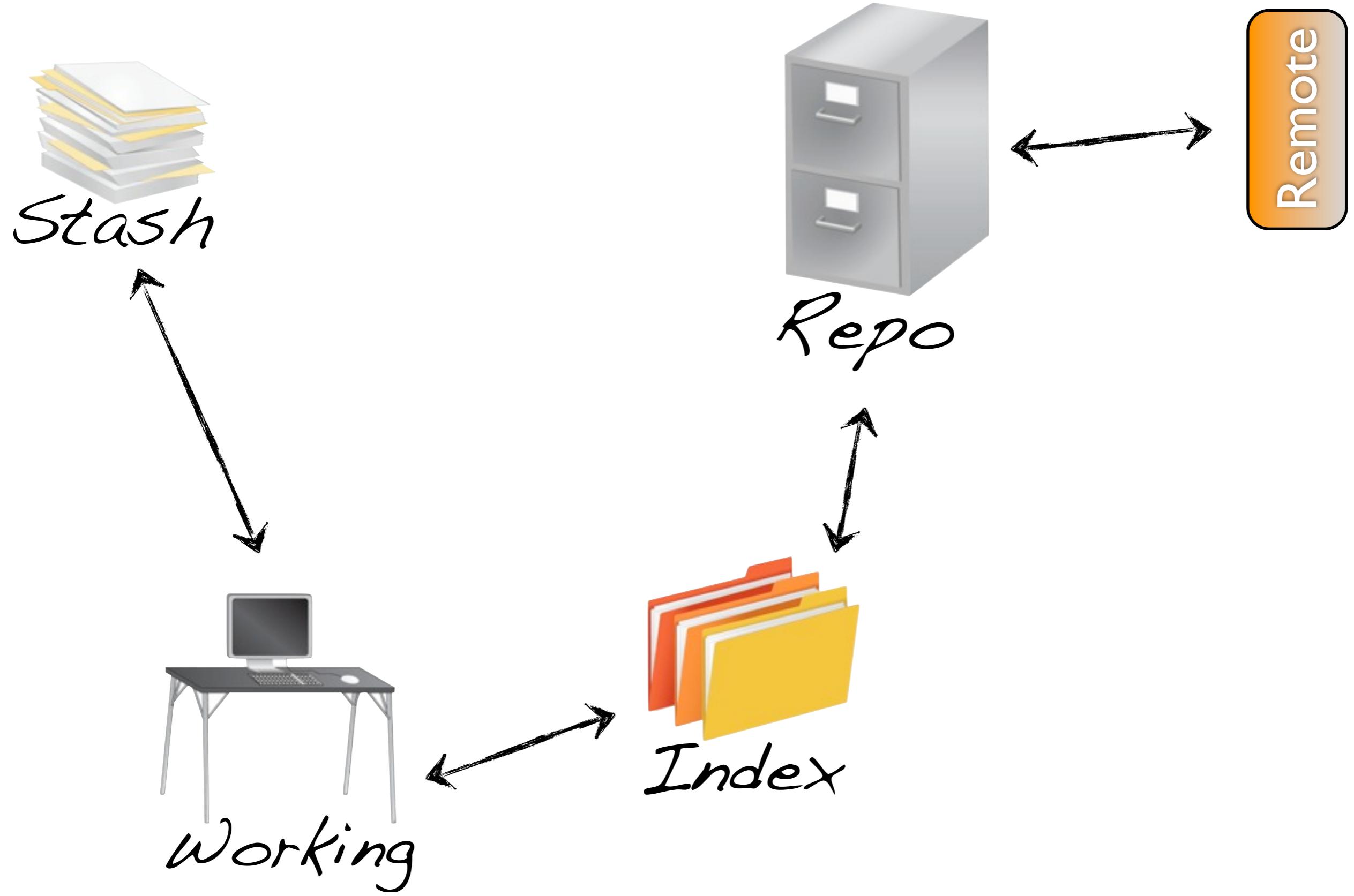


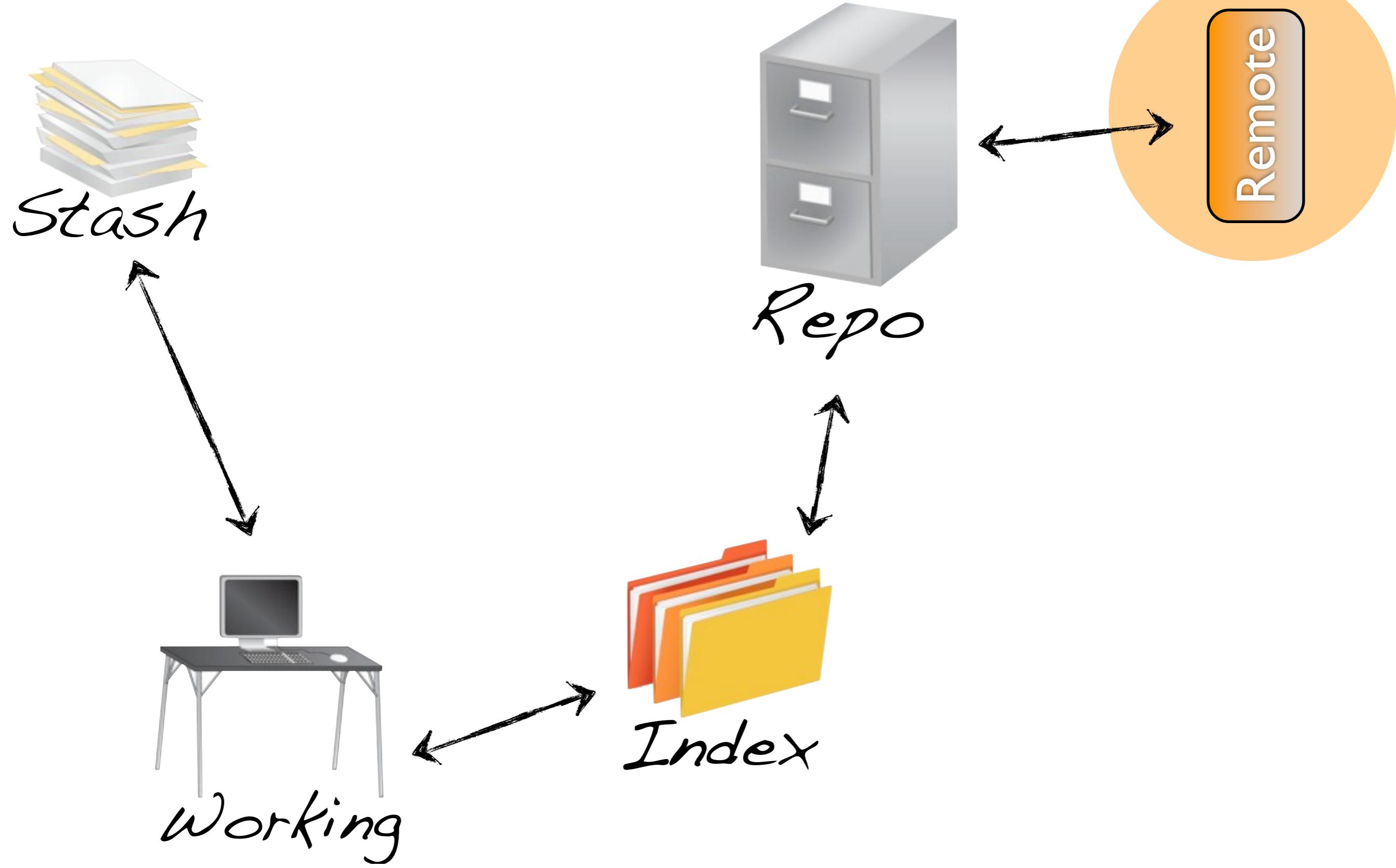
developer B

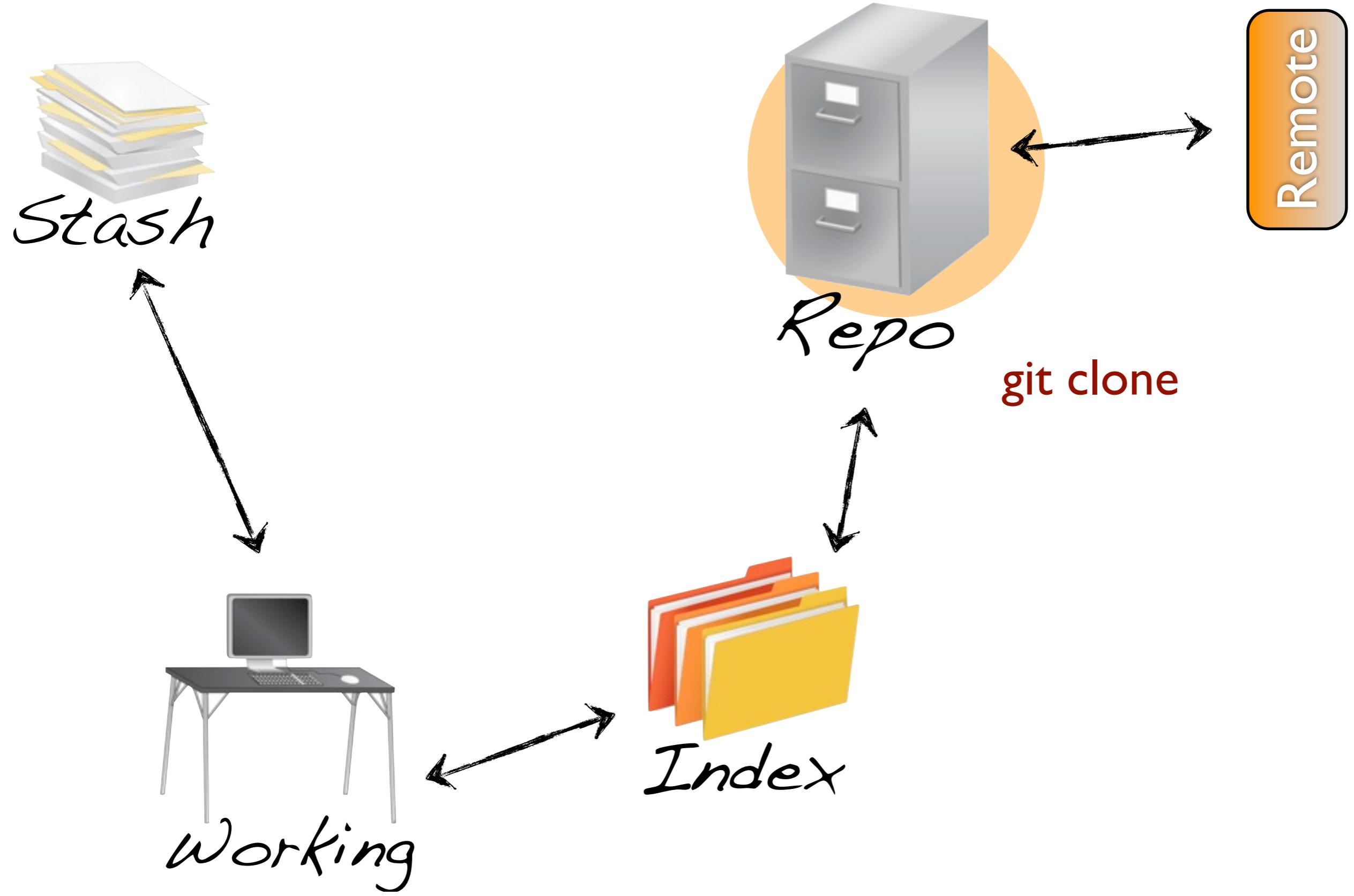


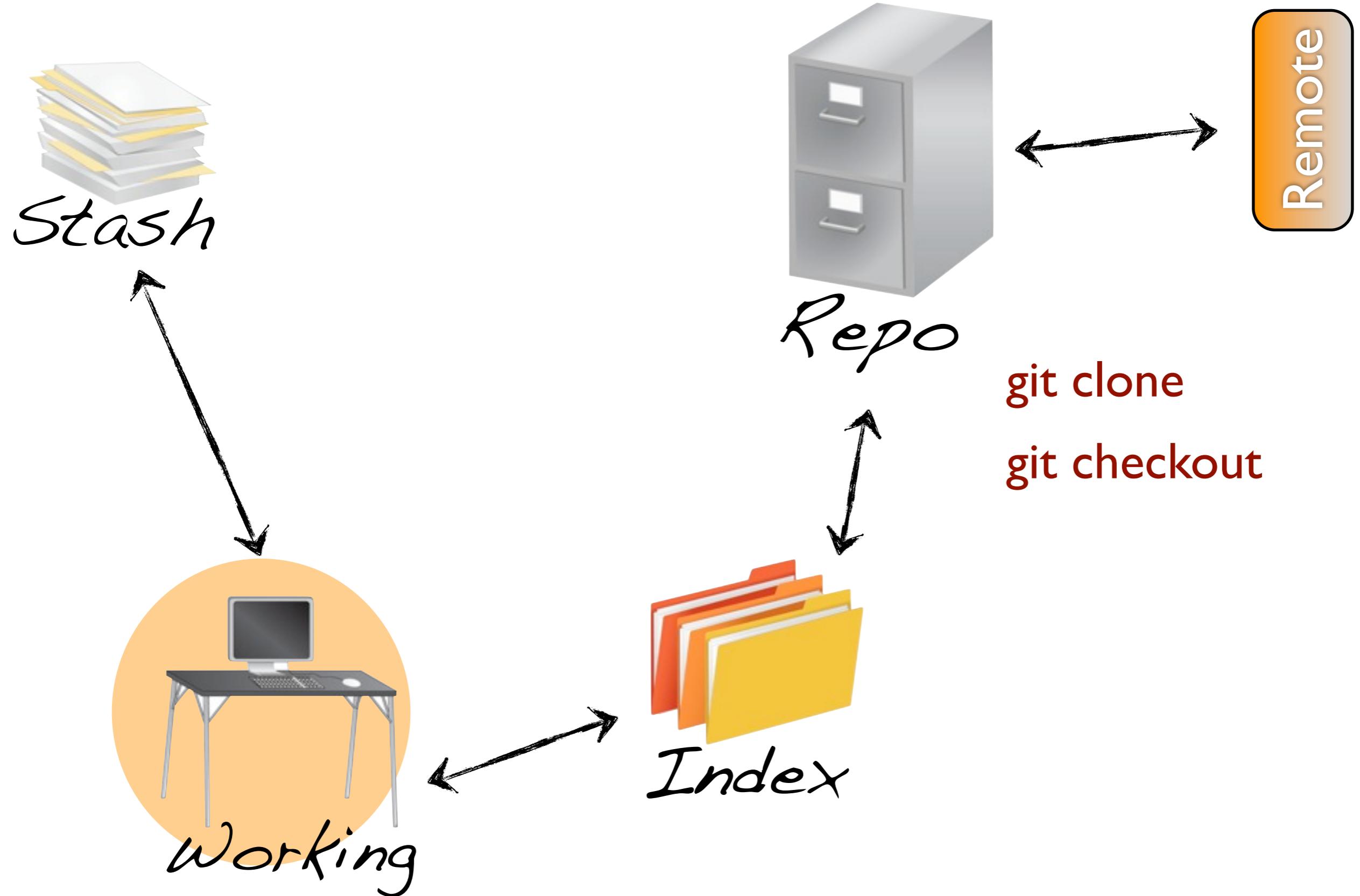
Three Stage Thinking

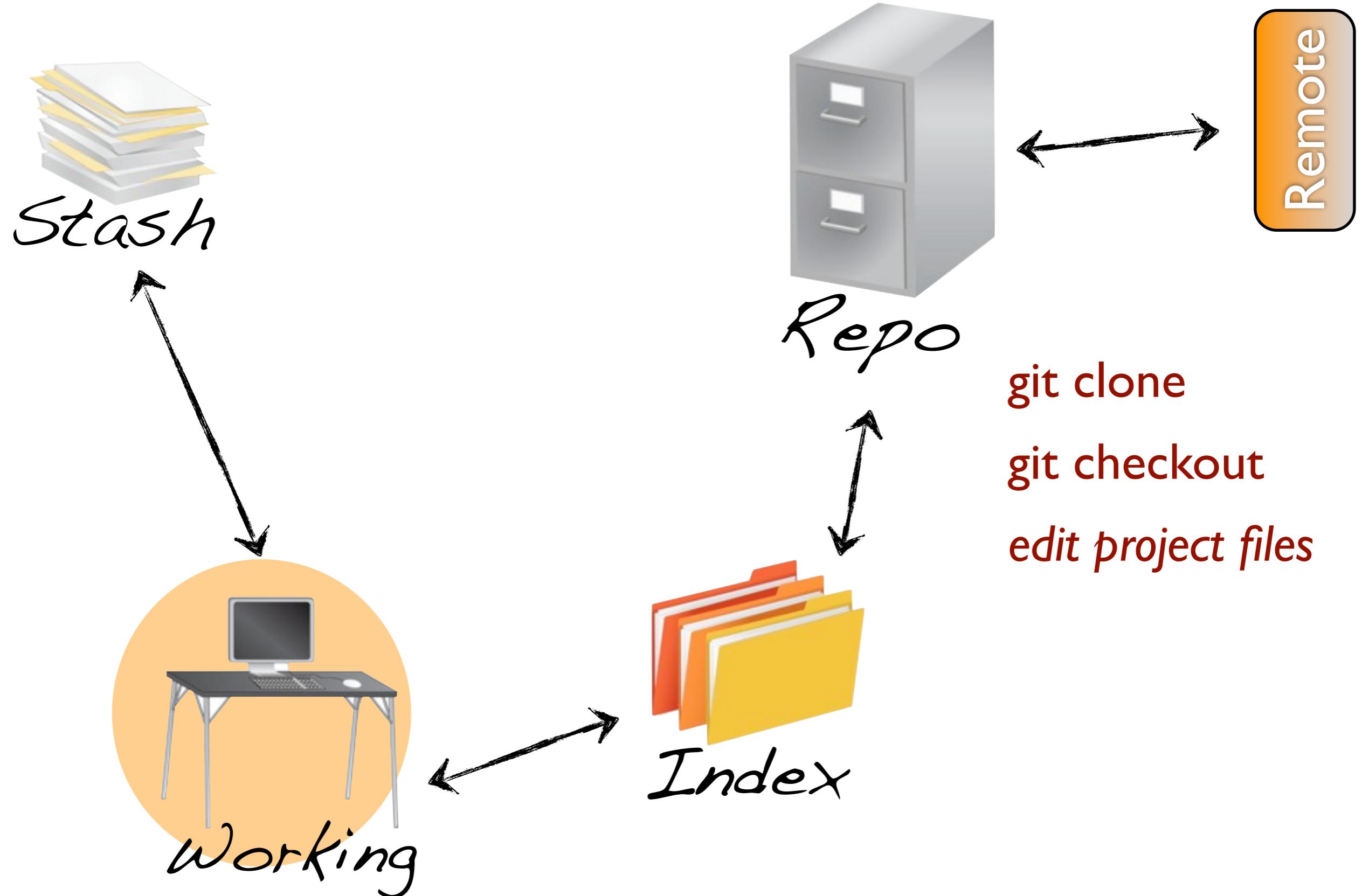


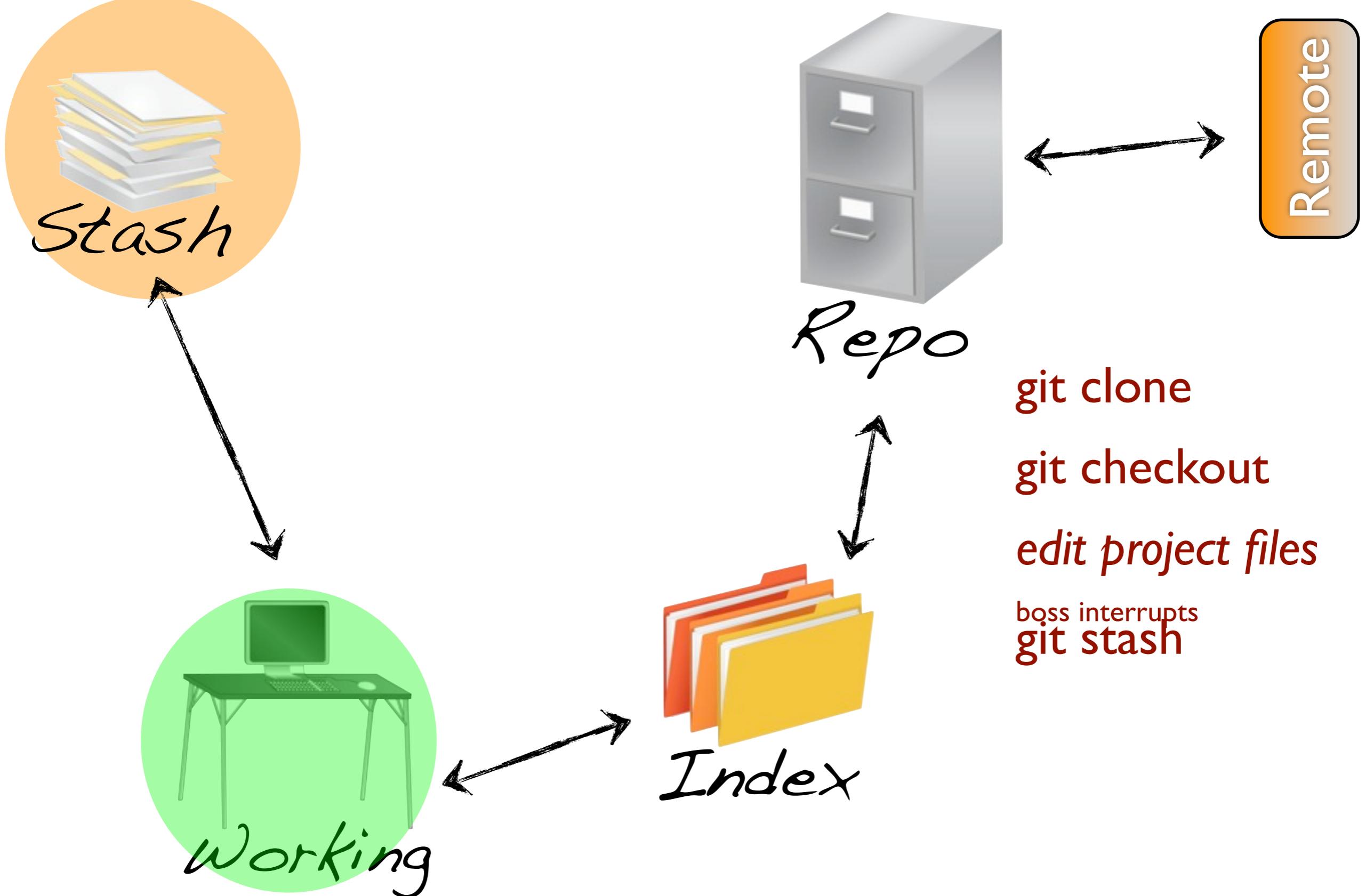


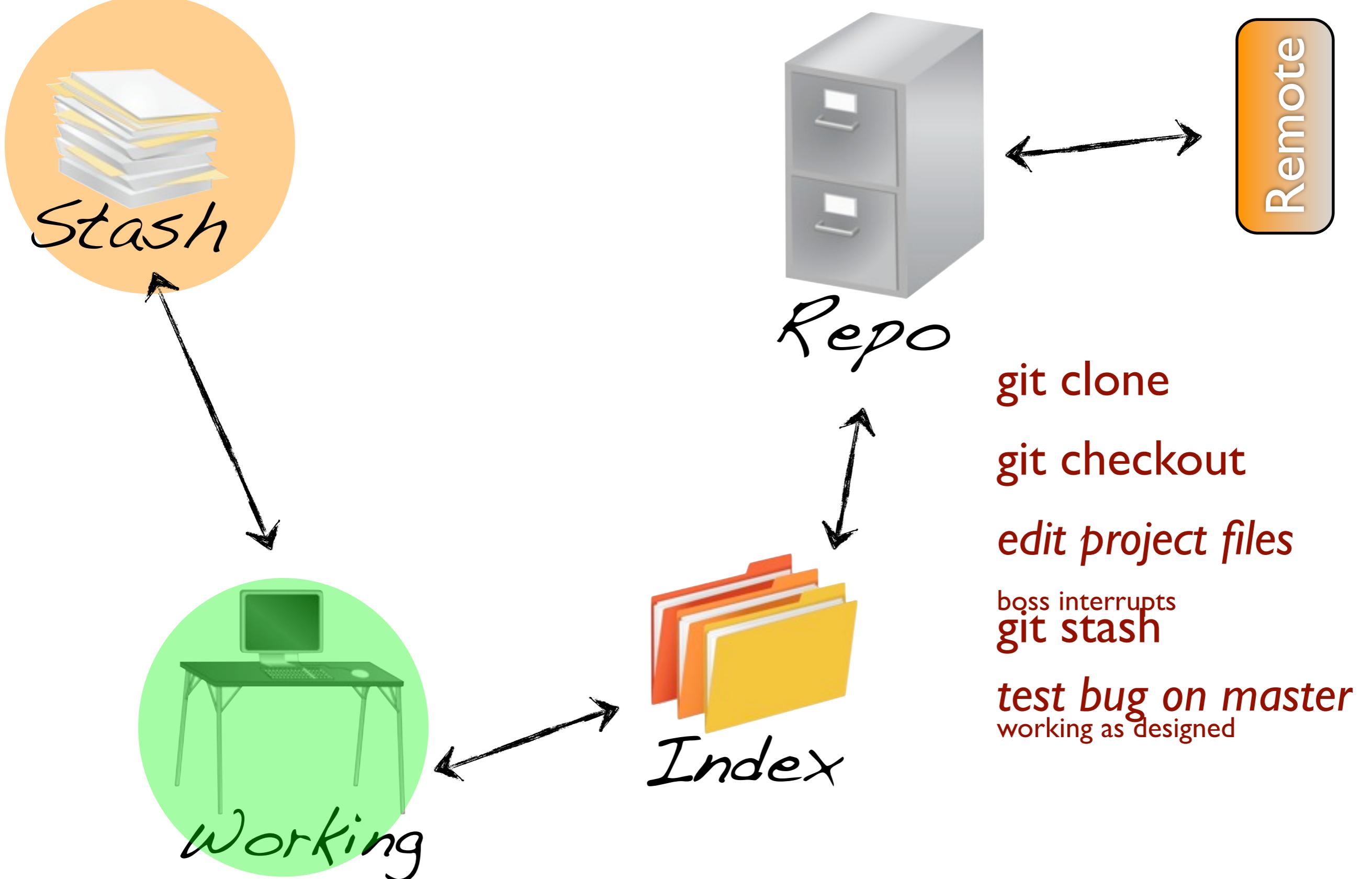


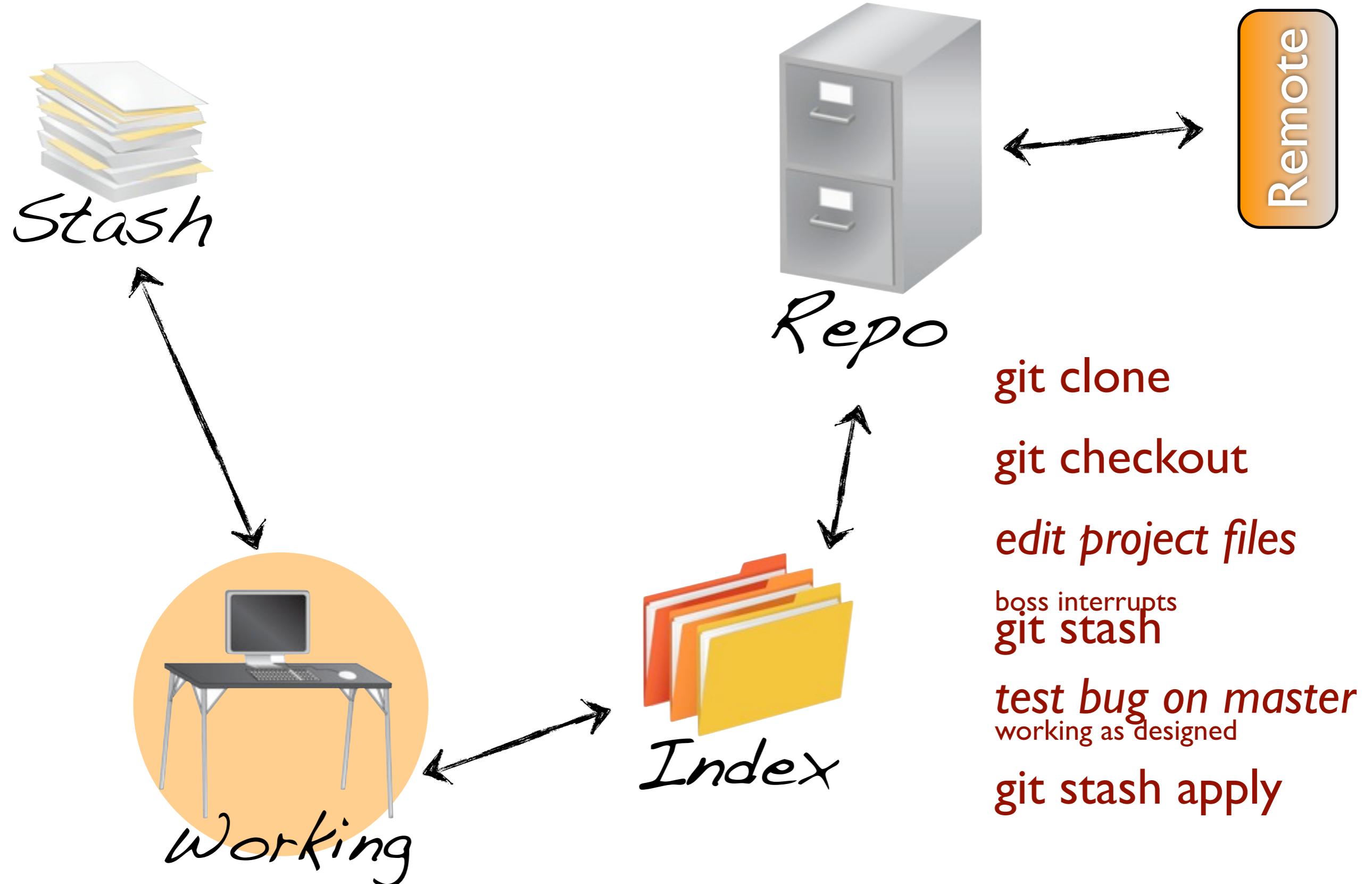


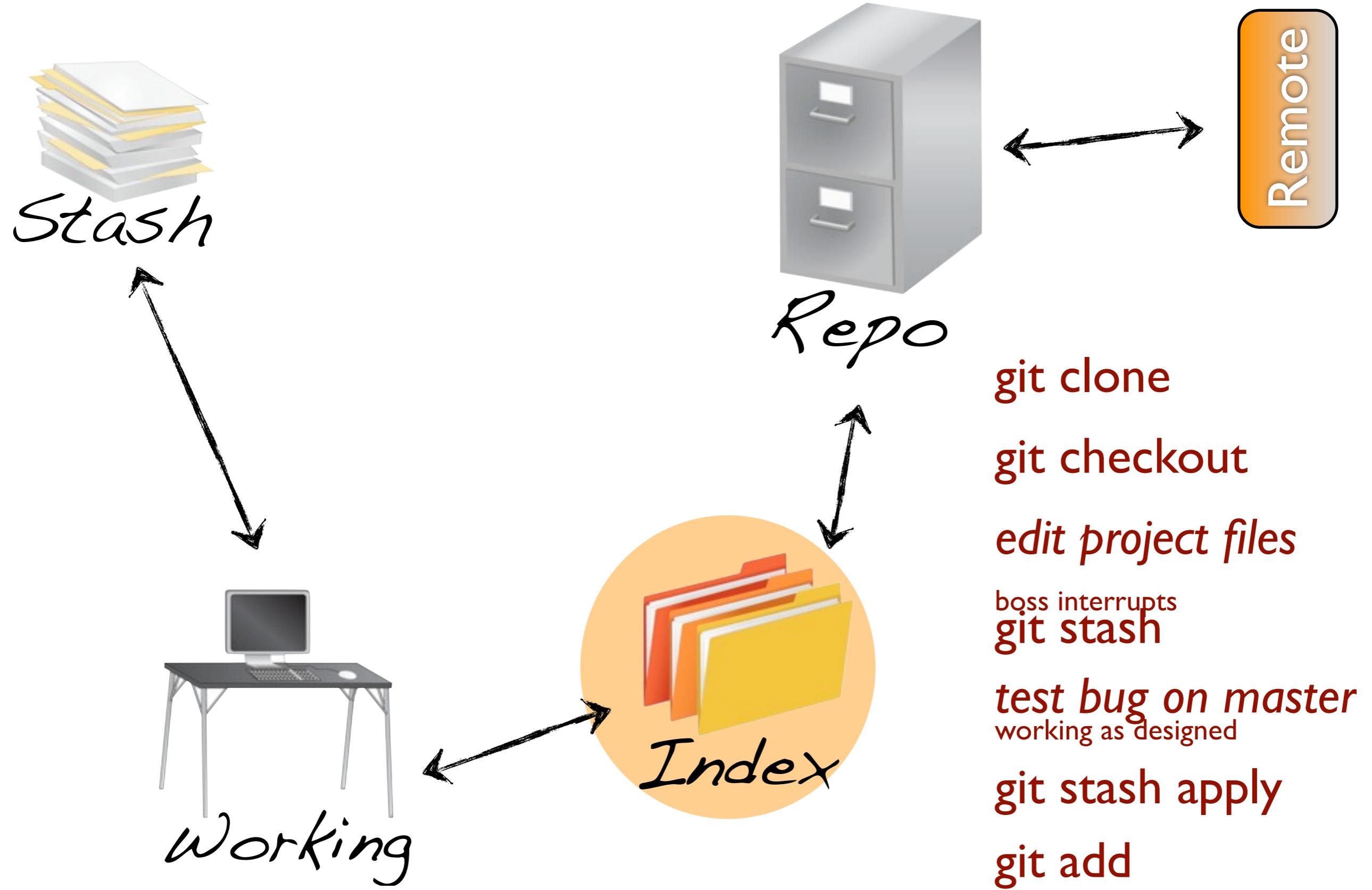


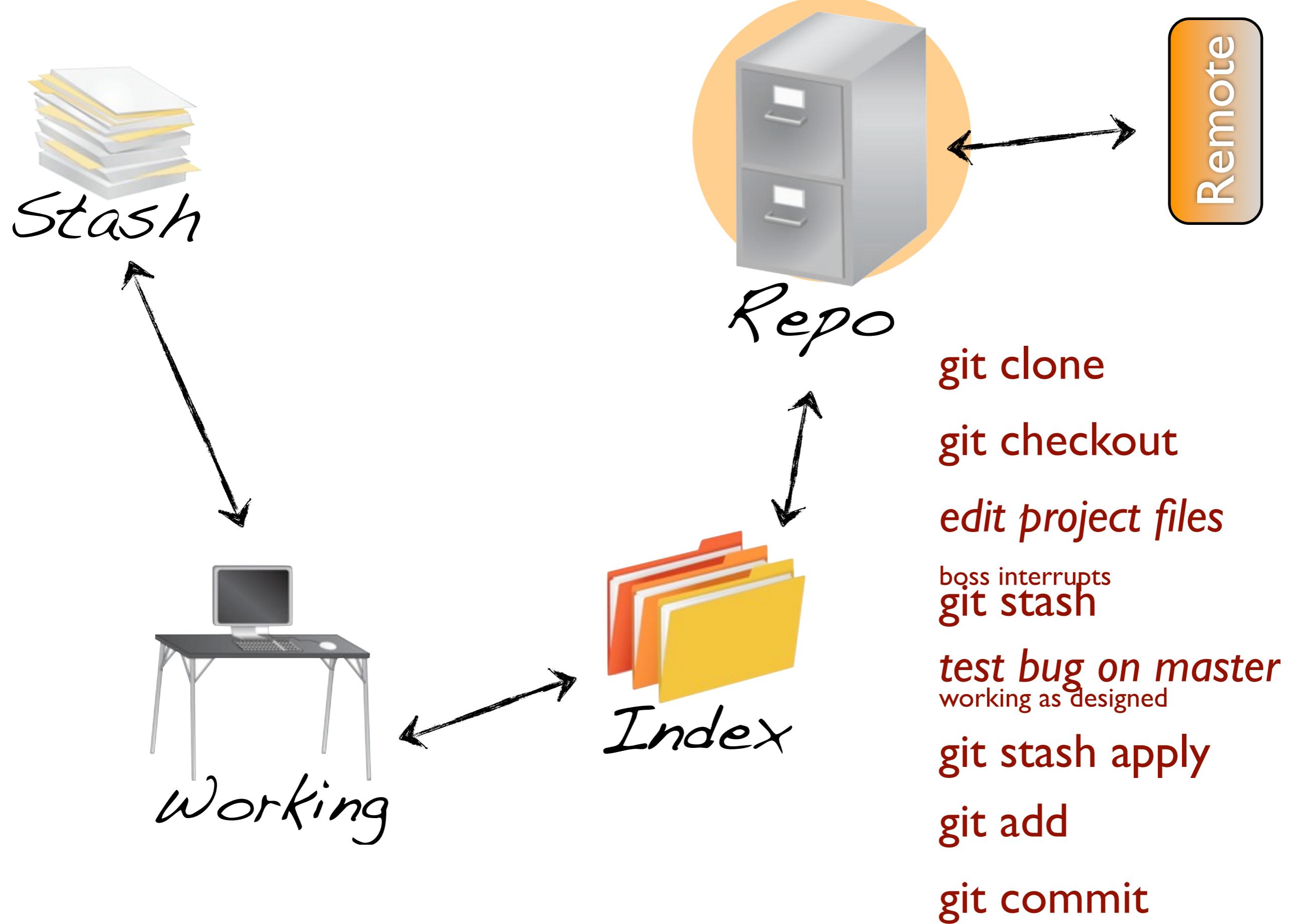


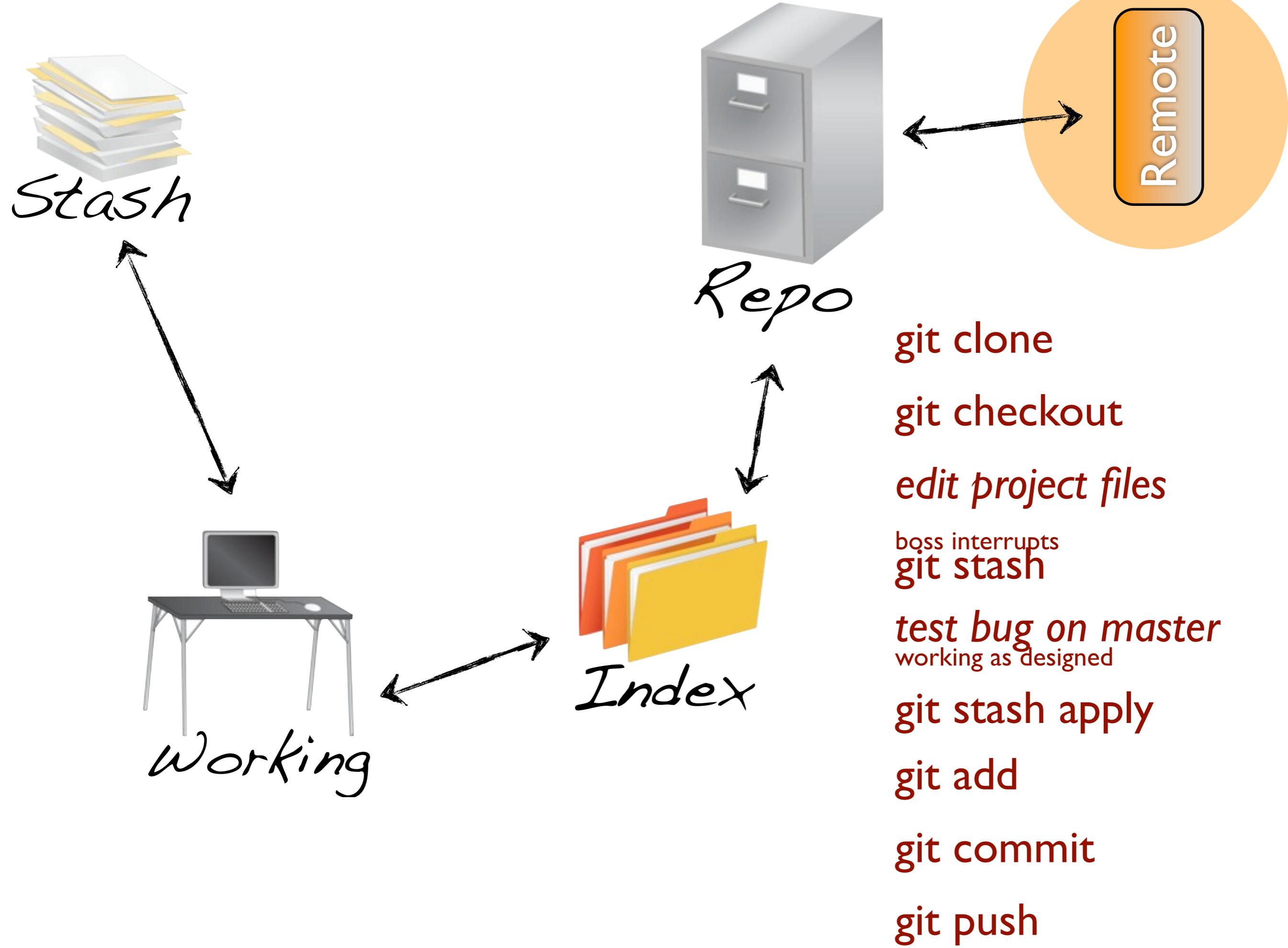












Working Offline

More available **connectivity**

More available connectivity



**More demand to work without
connectivity**



More demand to work **without**
connectivity

Offline Anything

Checkin

Offline Anything

Checkin

Add

Branch

List change log

Grep history

Rewrite history

Stash

Merge

Label

Remove

practically everything but push

Offline Anything

Checkin

Add

Branch

List change log

Grep history

Rewrite history

Stash

Merge

Label

Remove

~~practically everything but push~~





A tropical beach scene featuring palm trees, a clear blue sky, and two dark wooden lounge chairs on a sandy beach. A large, orange, hand-drawn style speech bubble is positioned in the upper right area of the image, containing the text.

Matthew: Cool!
Another commit to my
latest OSS project using
Git!

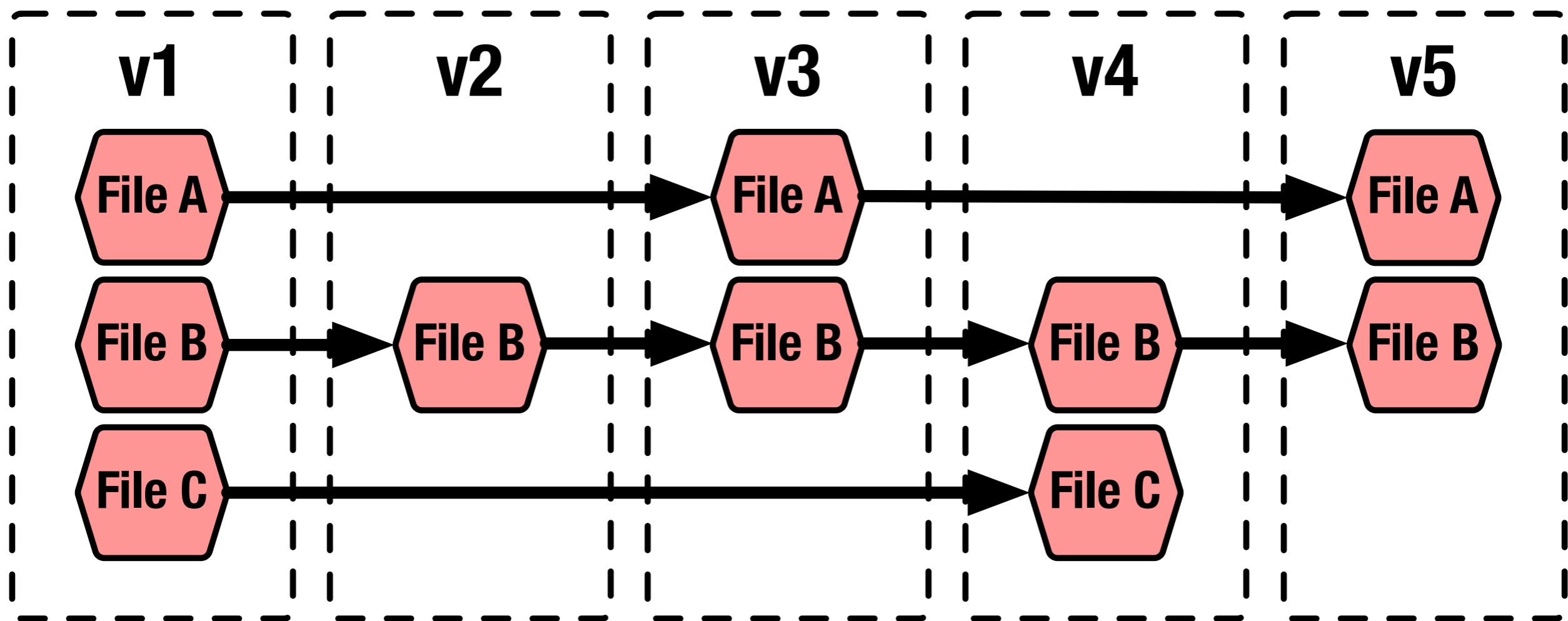


Matthew: Cool!
Another commit to my
latest OSS project using
Git!

Madelaine: I like
subversion
much better...

Storage Mechanics

Typical SCMs use deltas



cvs / Subversion / darcs / Mercurial



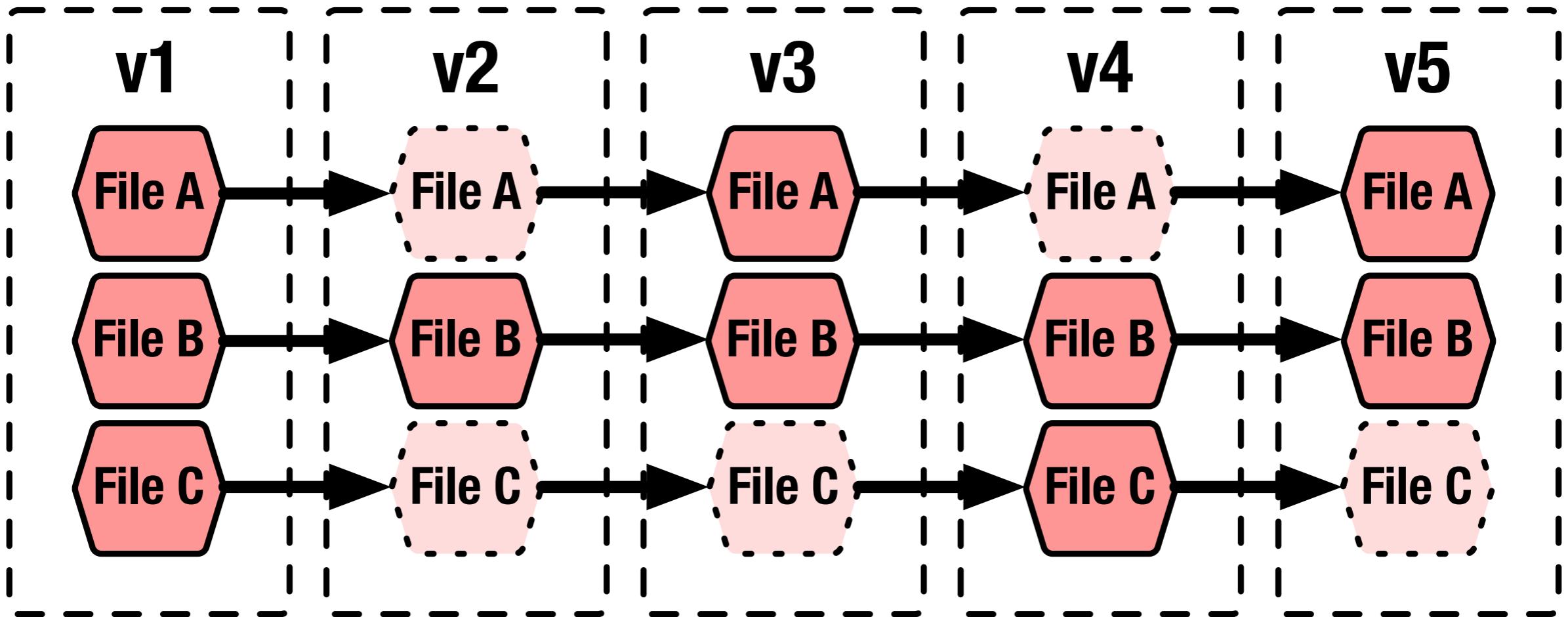
**Delta storage gets slower as the history
of a file gets longer**

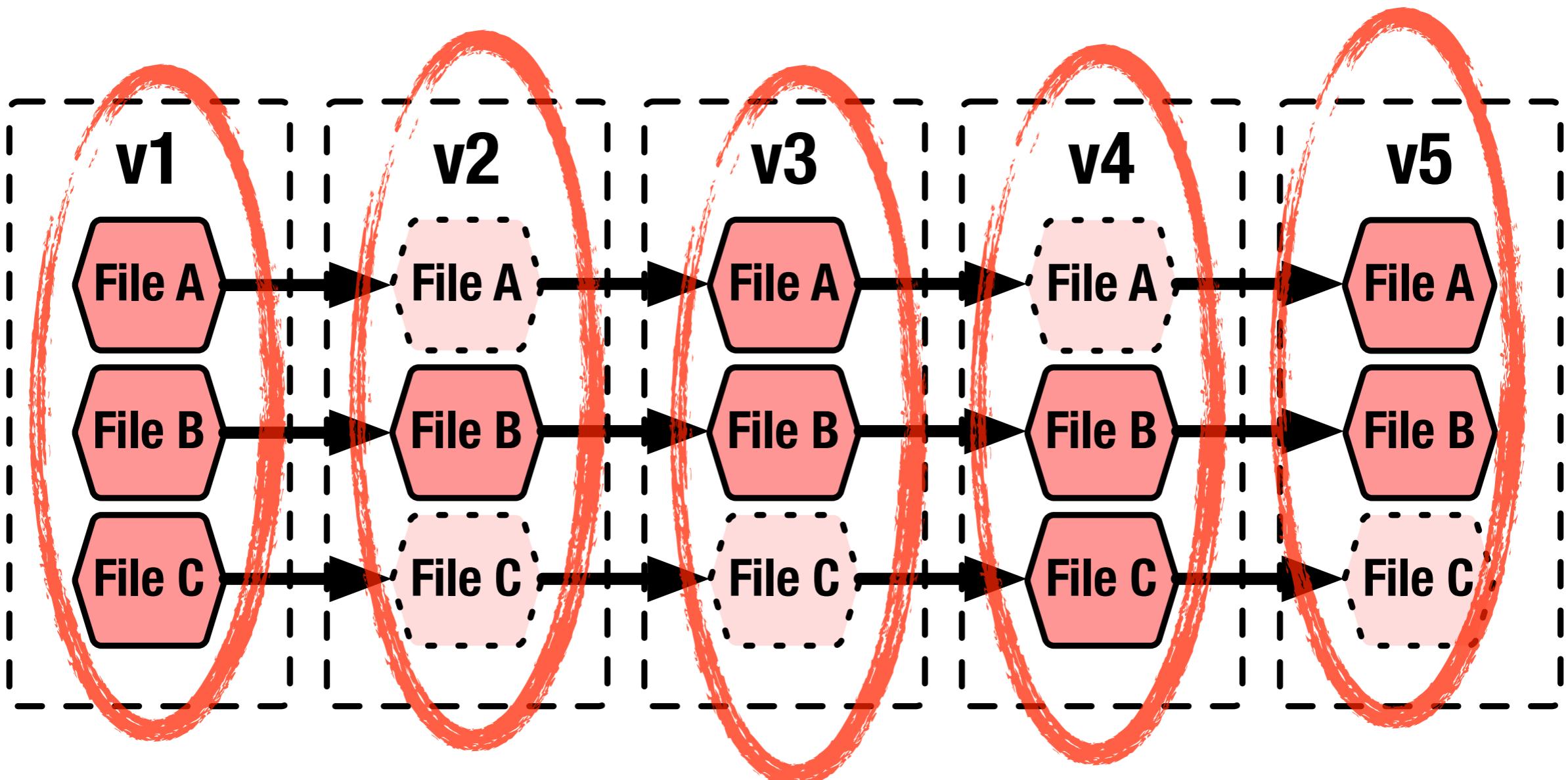
**Git uses Directed Acyclic Graph storage
(DAG)**

```
cp -r srcfolder srcfolder.prev
```

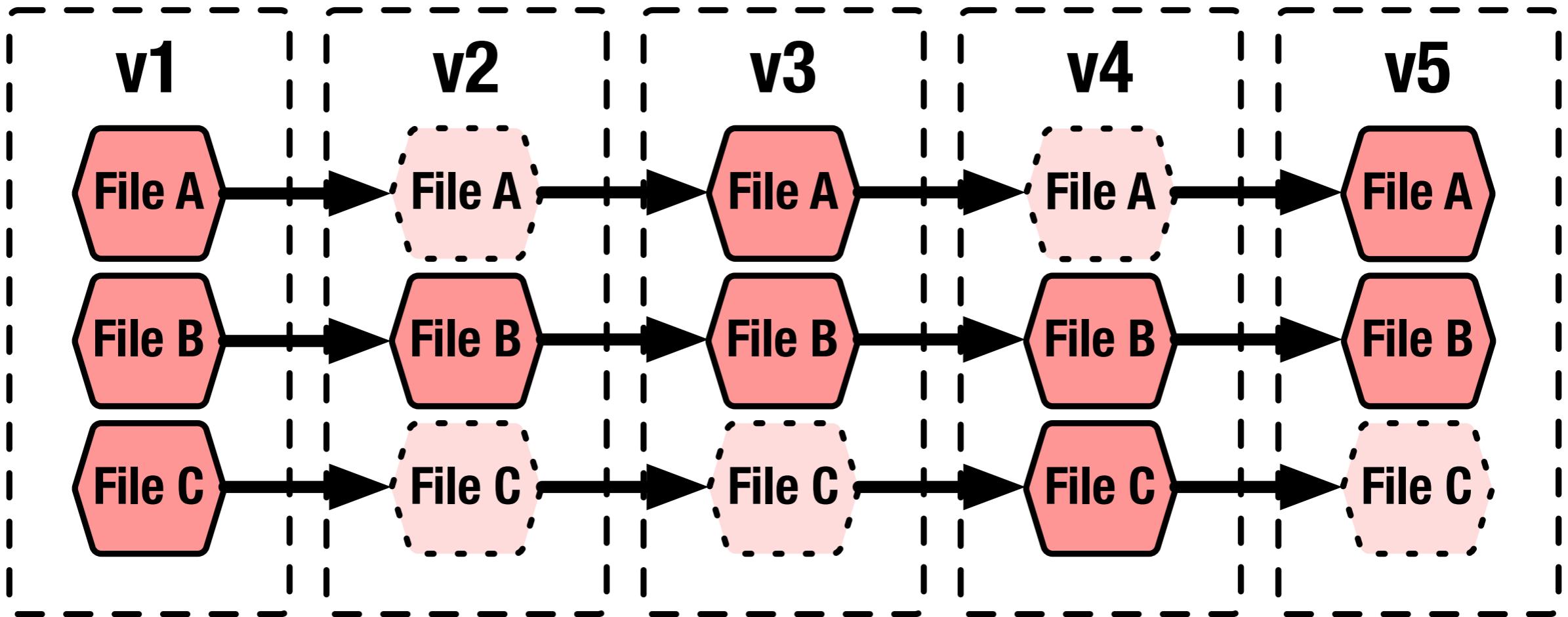
Copy of the entire tree per checkin

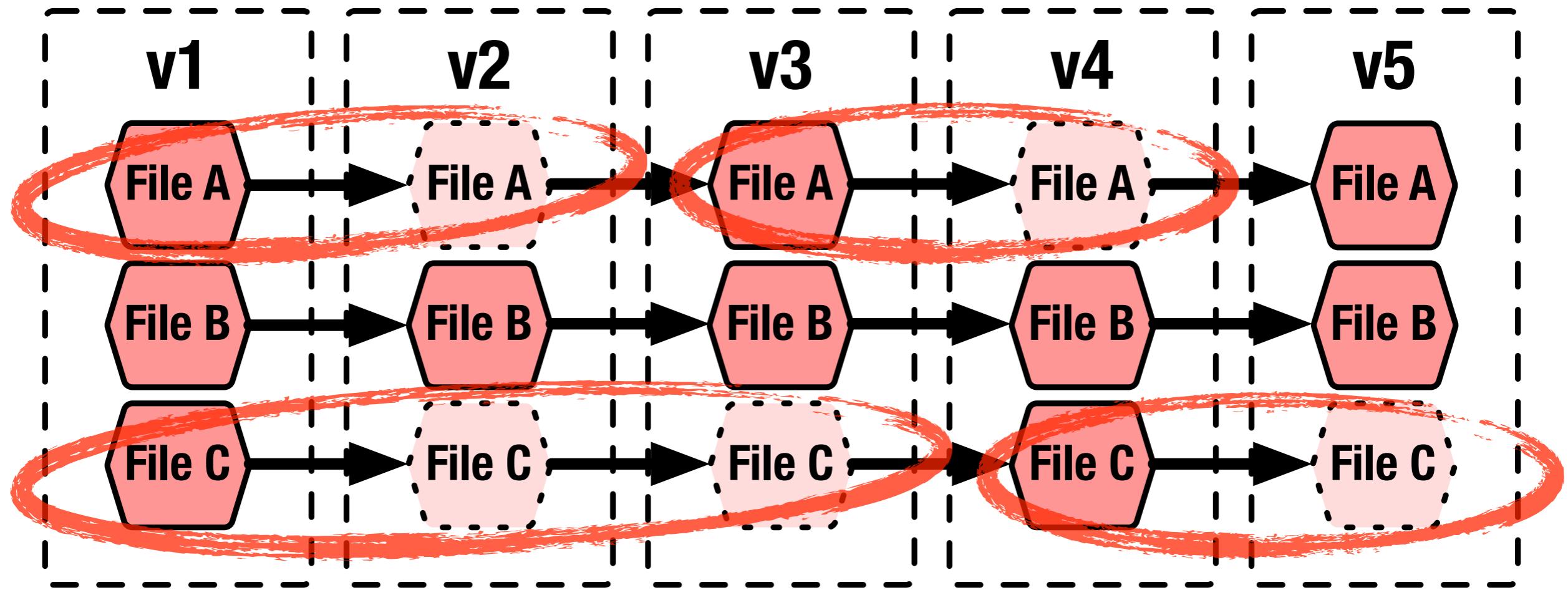




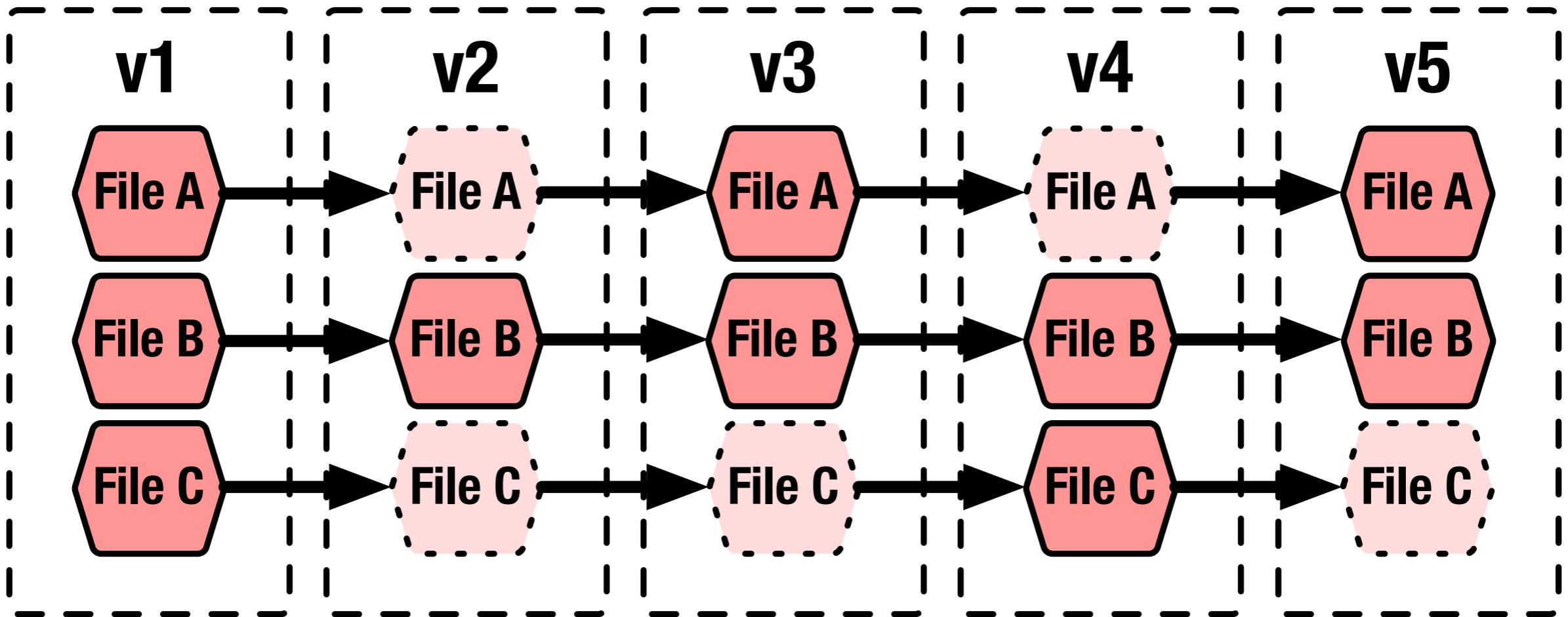


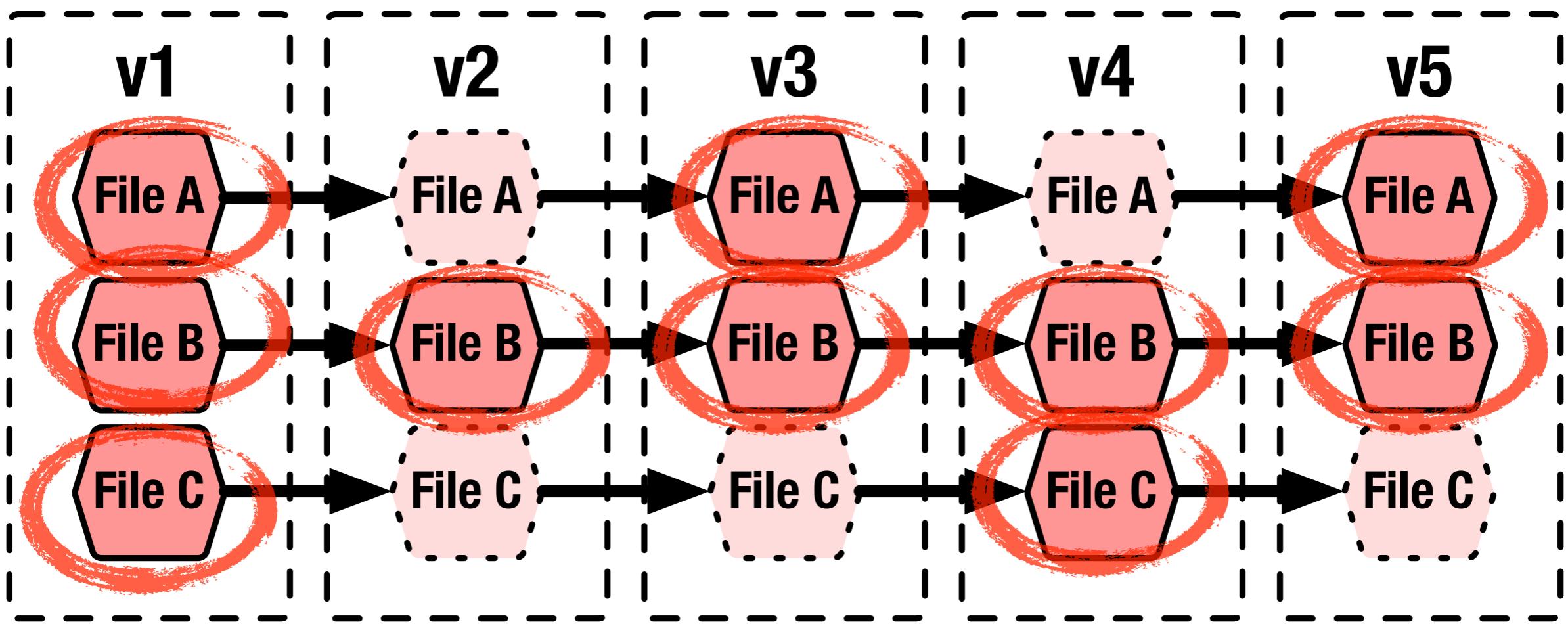
hardlinks to existing identical blobs



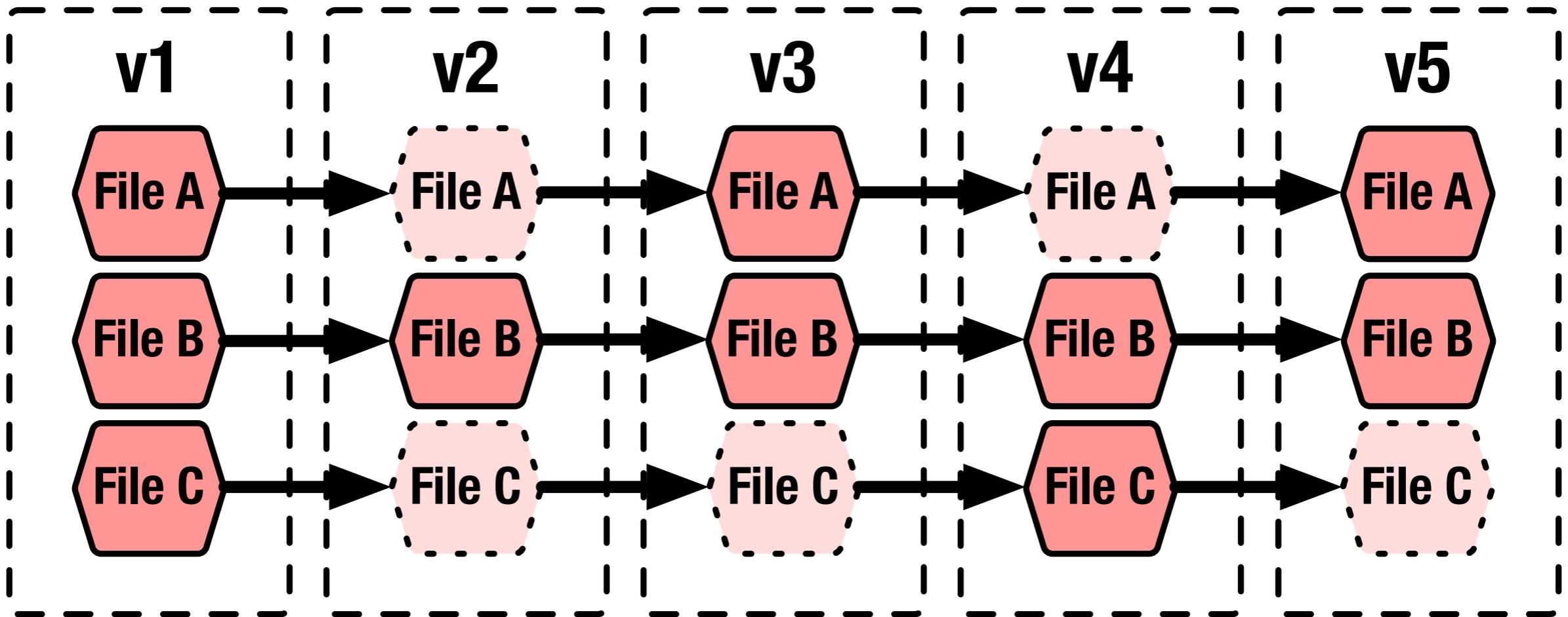


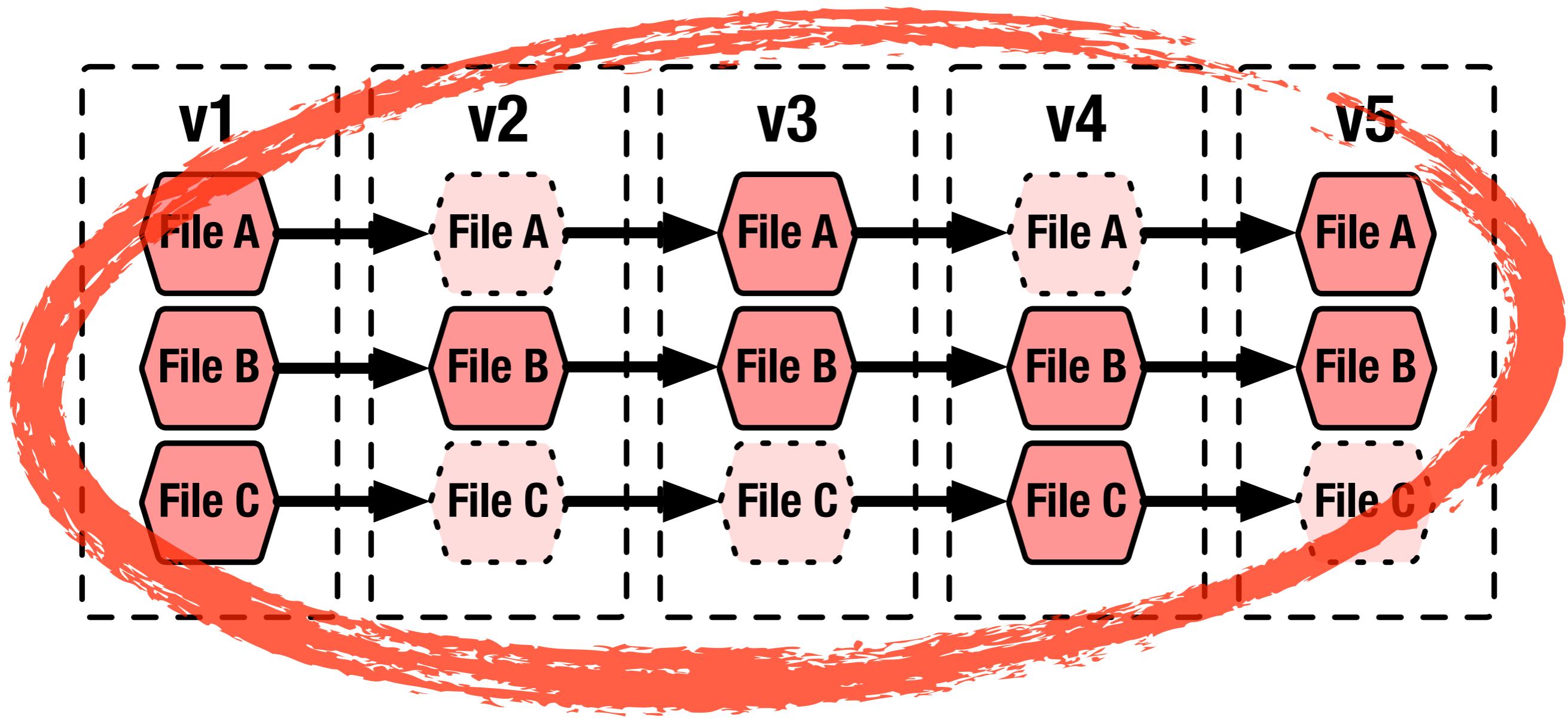
zlib deflates every blob at commit





zlib deflates entire repo



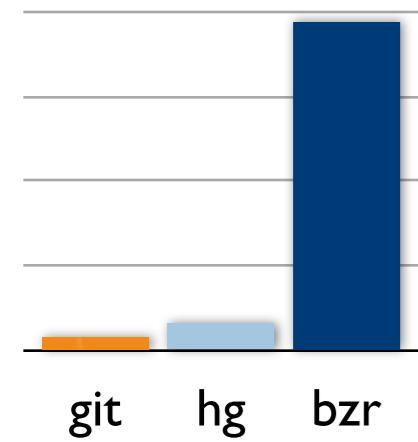
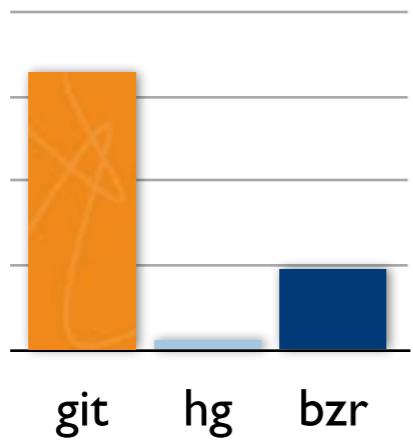
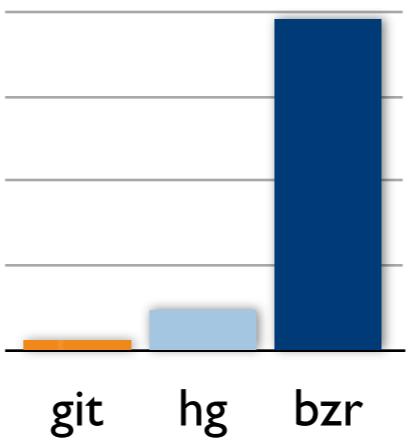
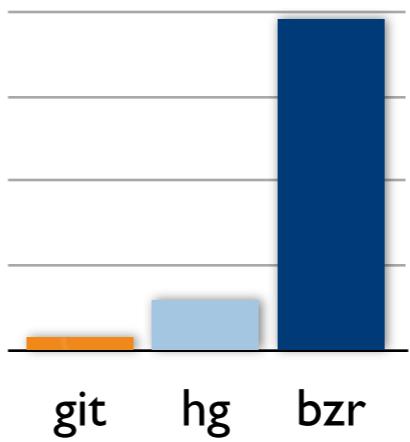
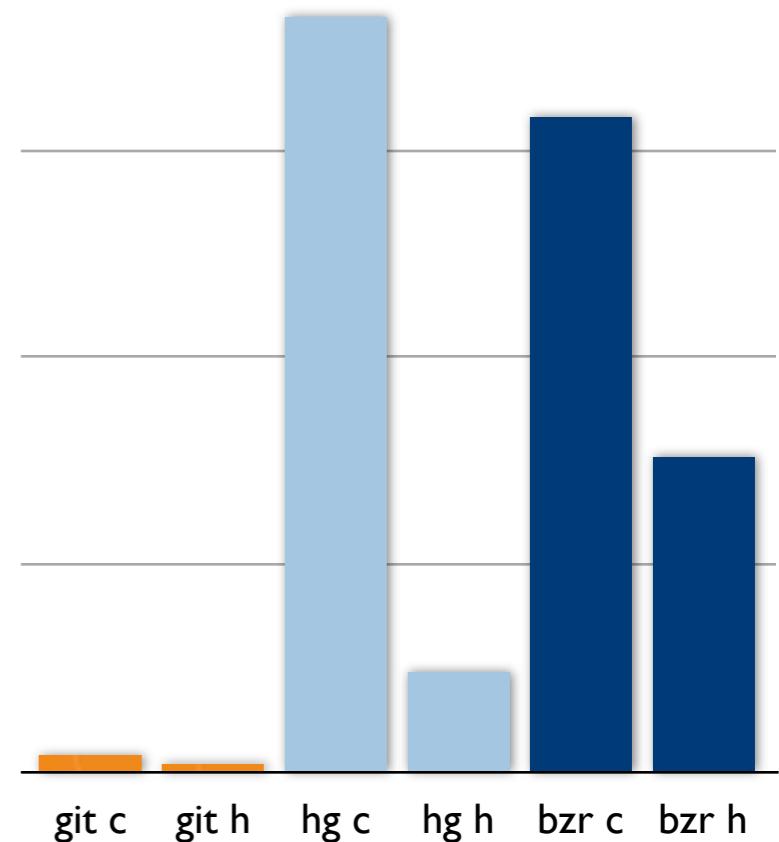
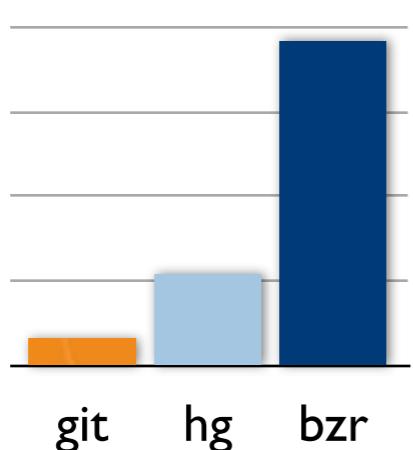
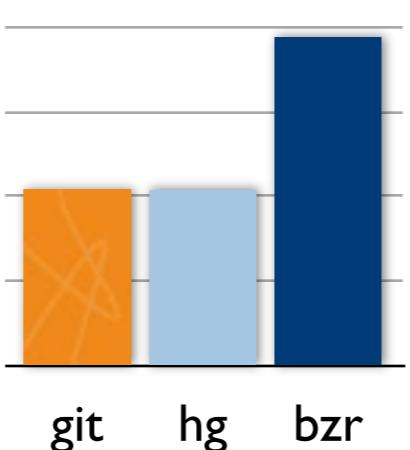


400MB Subversion repo
convert to a **70MB** Git repo

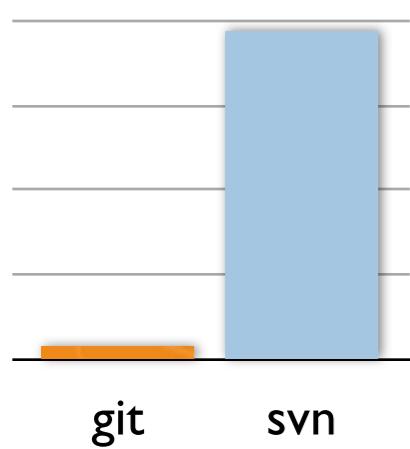


SPEED

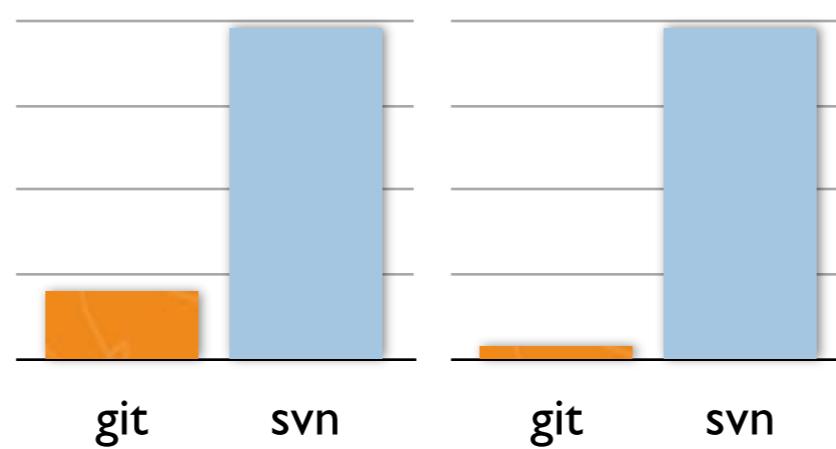
<http://whygitisbetterthanx.com>

Init**Add****Status****Diff****Branch (Cold/Hot)****Tag****Log****Commit (Lg)****Commit (Sm)**

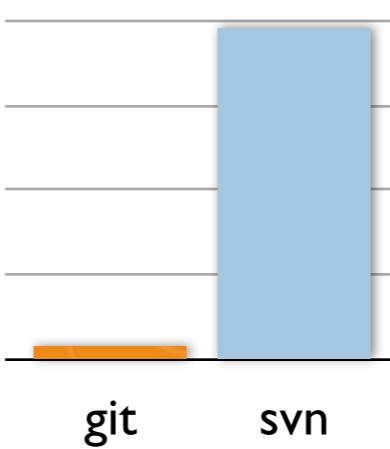
Init



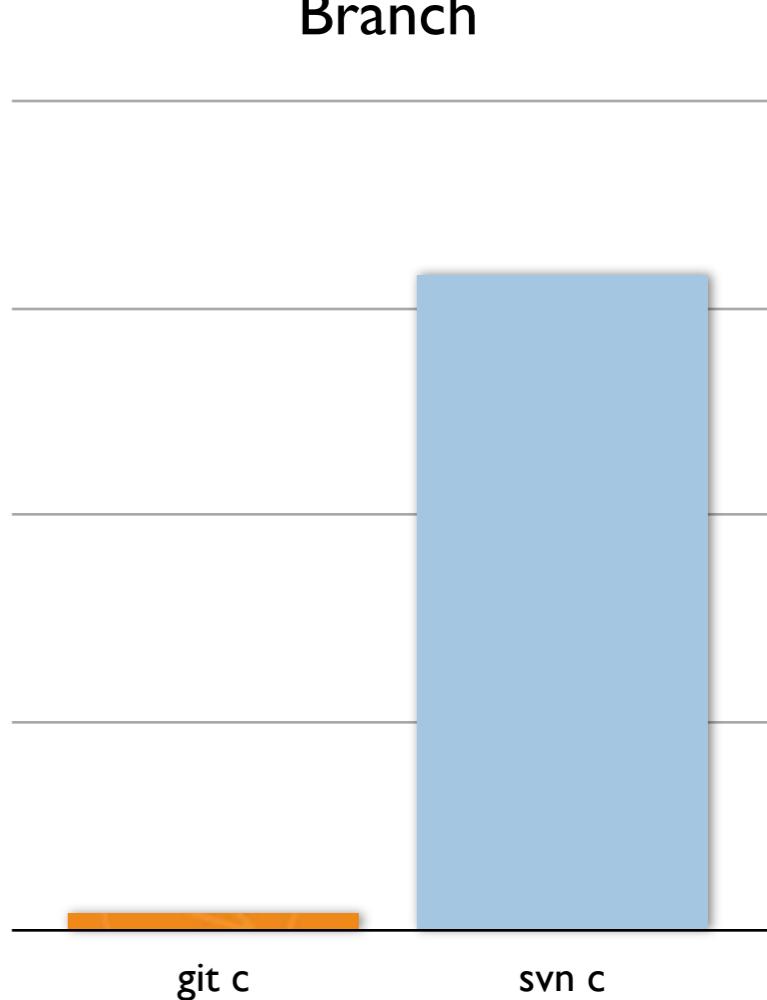
Status



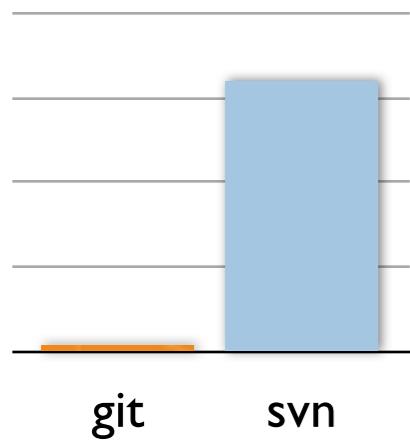
Diff



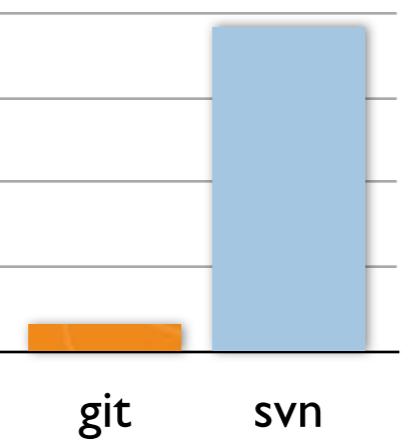
Branch



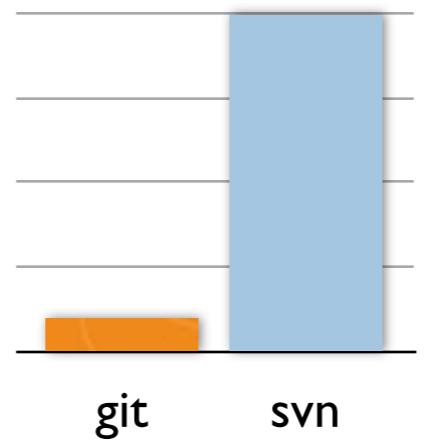
Tag



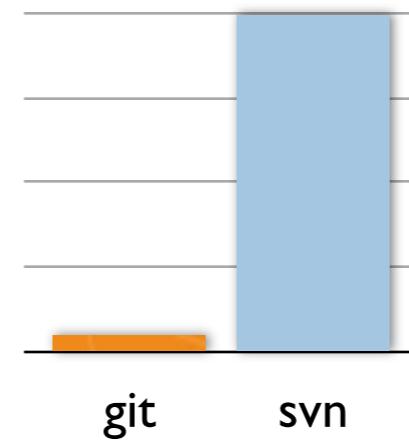
Log



Commit (Lg)



Commit (Sm)



10

TO

100 TIMES

FASTER

400MB Subversion repo
convert to a **70MB** Git repo

BRANCH, MERGE & REBASE

Branch

CHEAP BRANCHES

git branch newbranch

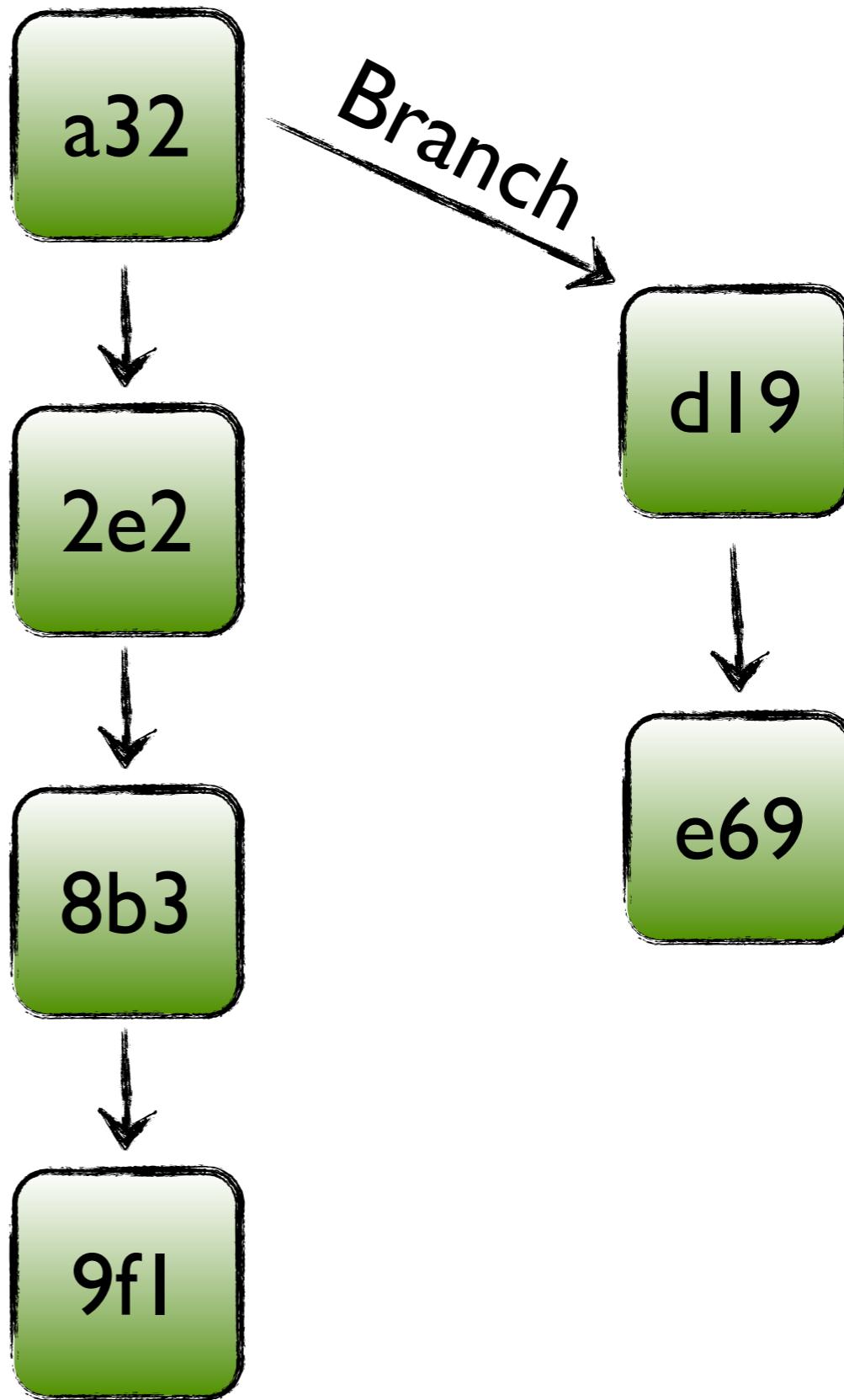
**STANDARD
BRANCH**

Master/Trunk/Main/Latest



STANDARD BRANCH

Master/Trunk/MainLatest



Merge

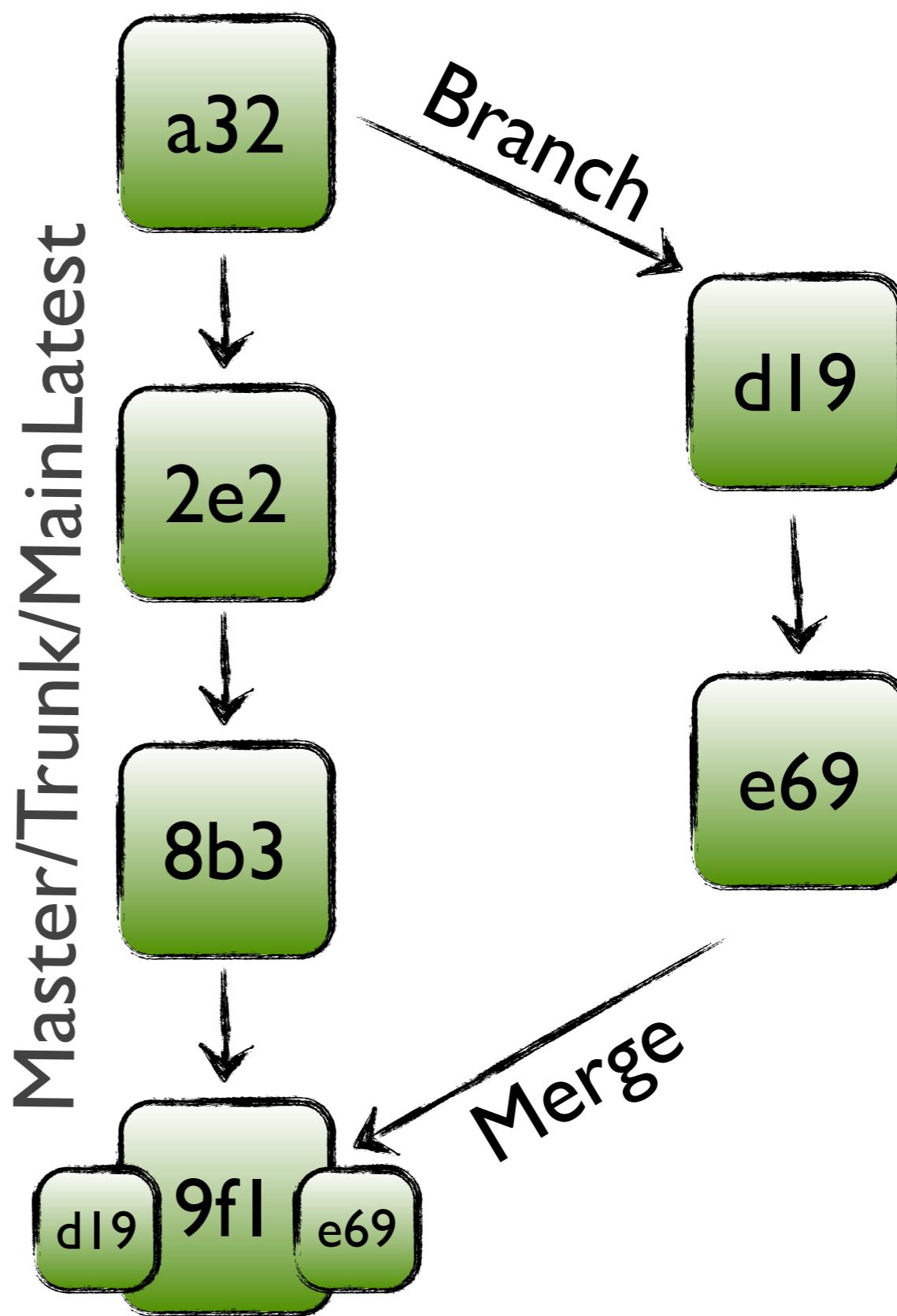
Merges lump branch changes into a
new unified commit

STANDARD MERGE

Master/Trunk/Main/Latest



STANDARD MERGE



MERGE Anything

```
git merge --no-commit <sourcebranch>
```

MERGE Anything

```
git merge --squash <sourcebranch>
```

MERGE Anything

```
git merge <sourcebranch1> <sourcebranch2>
```

Rebase

Not a merge

simulates team members **taking turns** working
(one person at a time)



REBASE for Clarity

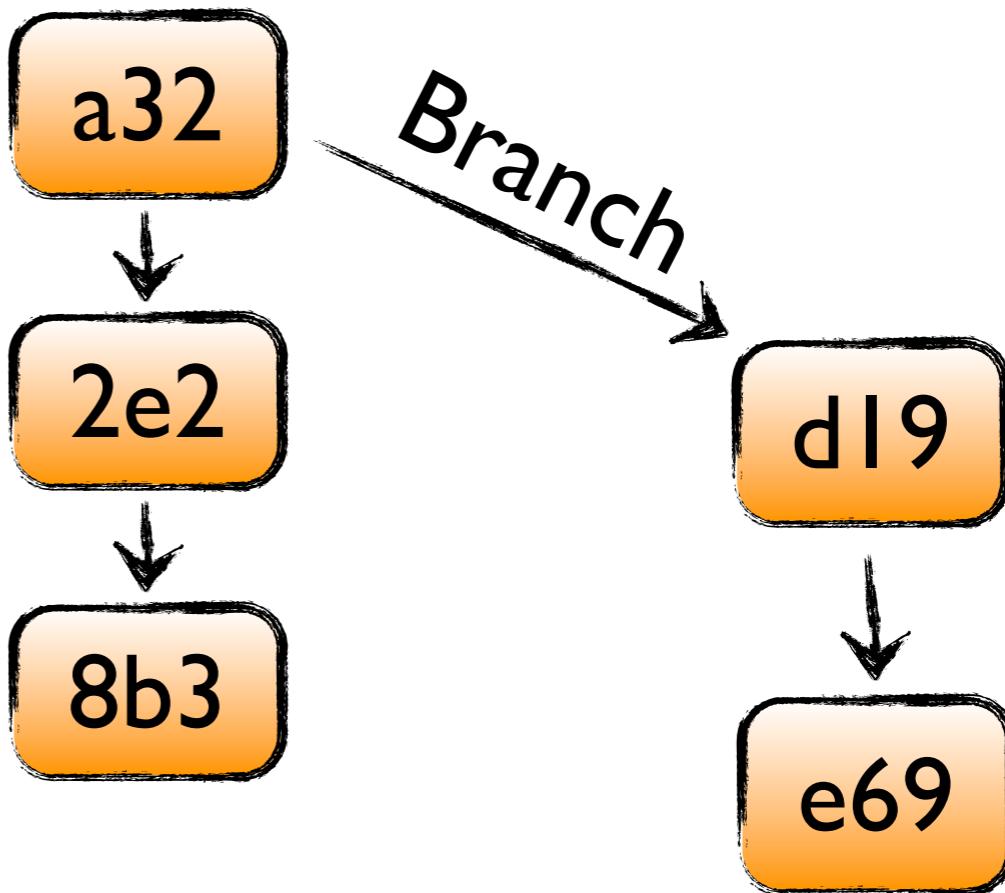
```
git checkout myfeaturebranch  
git rebase master
```

REBASING

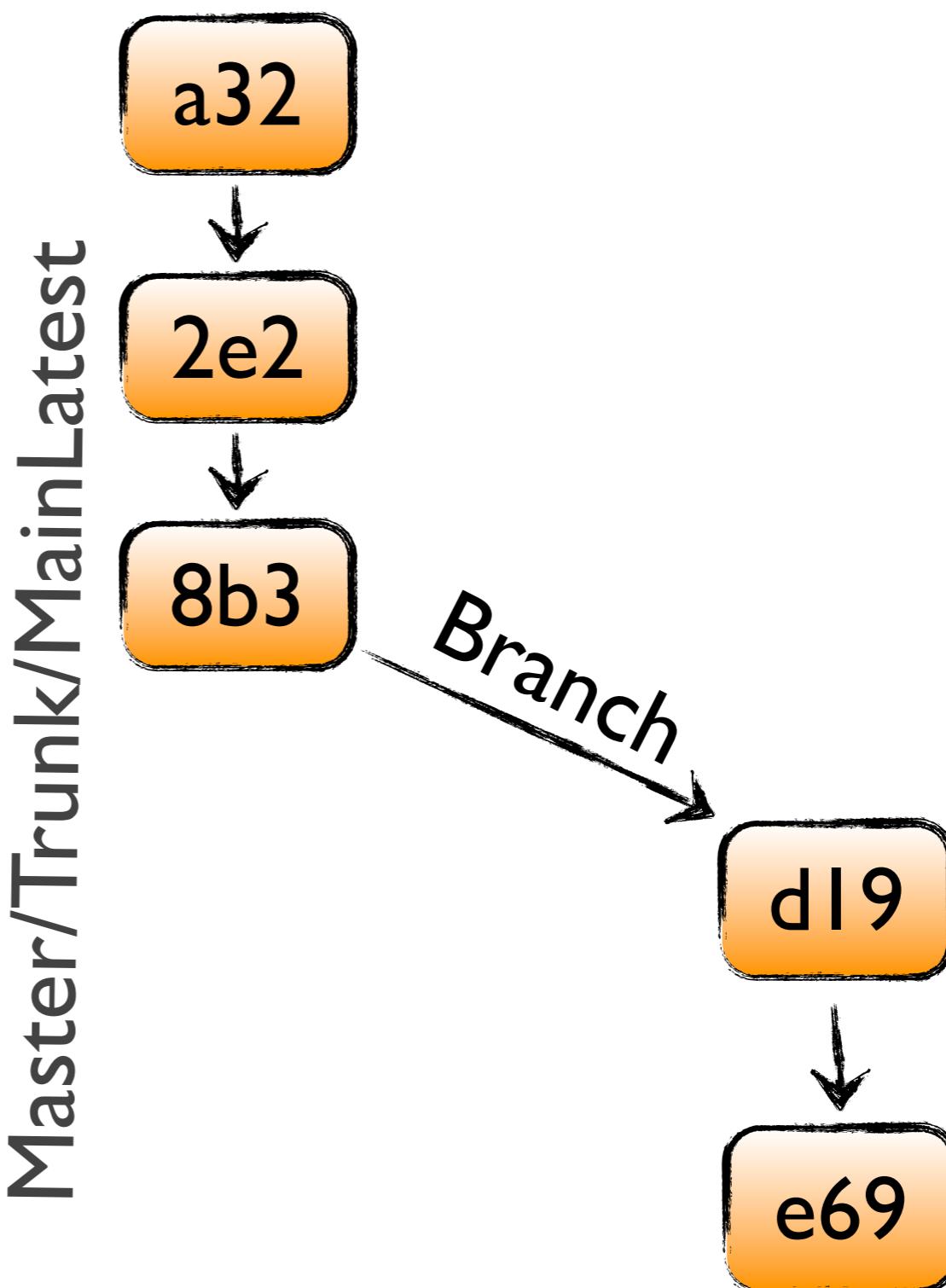
Master/Trunk/MainLatest

REBASING

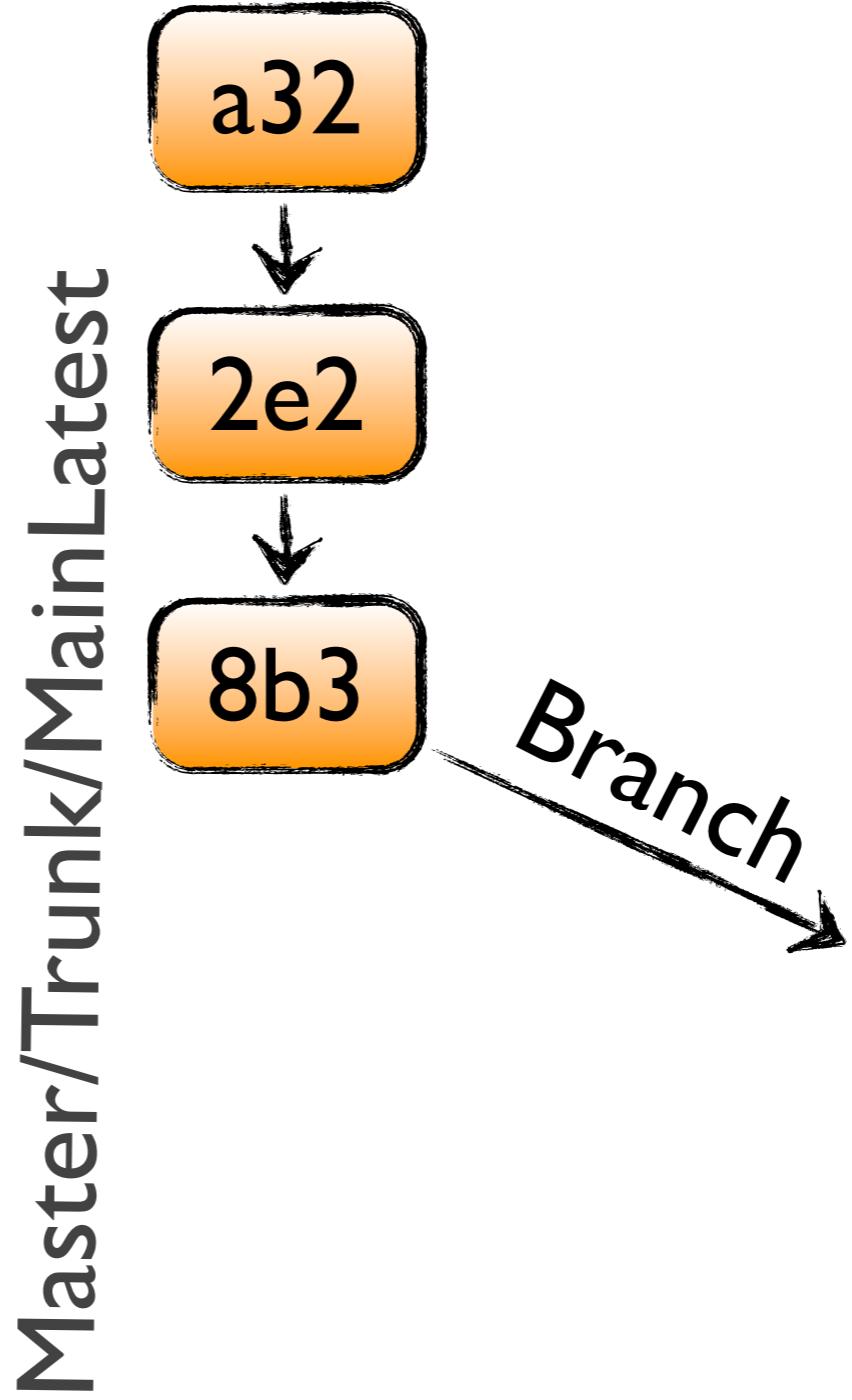
Master/Trunk/MainLatest

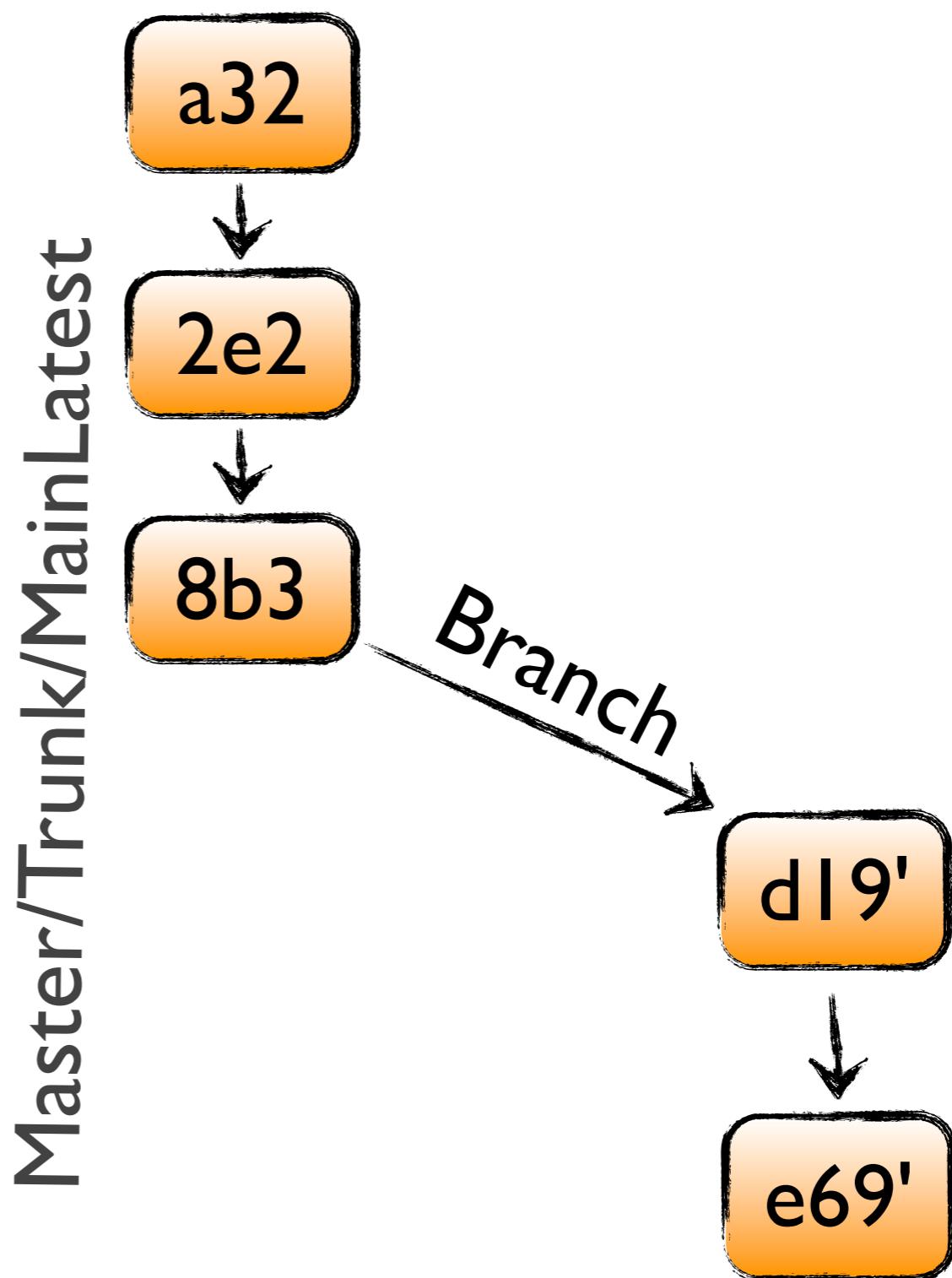


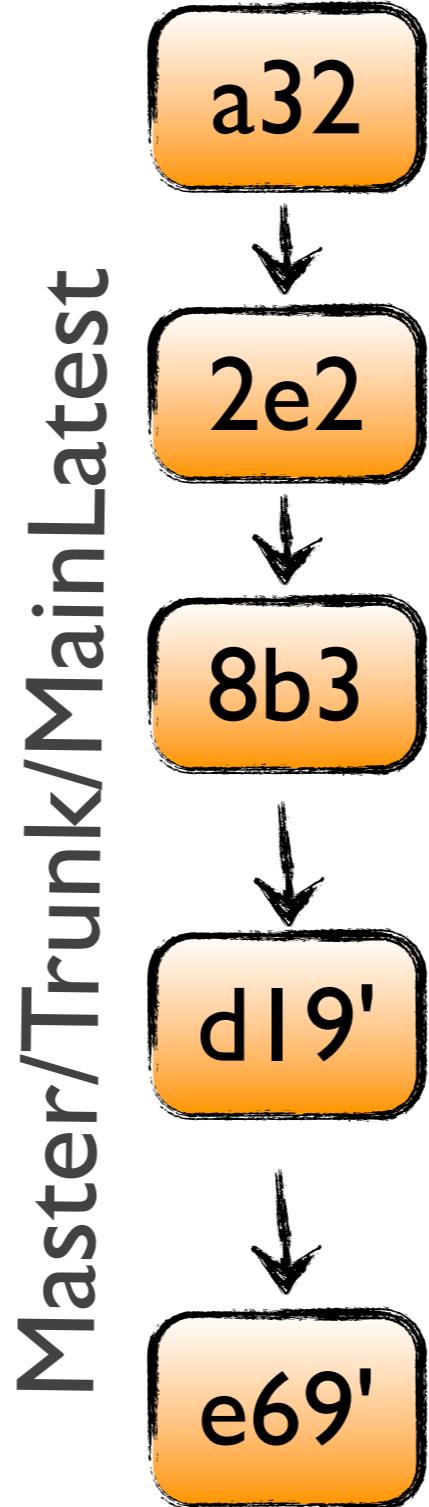
Rebase



Master/Trunk/MainLatest







Who's Got Git?

ANDROID

open source project

Work with the code

[Get source](#)

[Report bugs](#)

[Contribute](#)

[Code reviews](#)

[Patch history](#)

[Known issues](#)

[Mailing lists](#)

[Documentation](#)

About the project

[Project layout](#)

[Release features](#)

[Roadmap](#)

[Project roles](#)

[Licenses](#)

Develop applications

If you are interested in developing applications for Android, visit the [Android application developer site](#).

About Android

For general information, visit [android.com](#).

Get source

For information about current problems and fixes, see [Known issues](#).

This document describes how to set up your local work environment, how to use Repo to get the Android files, and how to build the files on your machine.

Related reading:

- For an overview of the entire code-review and code-update process, see [Workflow](#).
- For reference details about Repo, see [Using Repo and Git](#).

What's in the source?

For a description of all the projects that make up the Android source code, see [Project layout](#). To see snapshots and histories of the files available in the public Android repositories, visit the [GitWeb](#) web interface.

The source is approximately 2.1GB in size. You will need 6GB free to complete the build.

Setting up your machine

To build the Android source files, you will need to use Linux or Mac OS. Building under Windows is not currently supported.

Linux



Search:

search is working now...

Job Board

[Website](#)

[Development](#)

[Enterprises is looking for a Rails Architect.](#) See more on the [Job Board](#).

Worth attention

[Workshops](#)

[Podcast](#)

Categories

Rails is moving from SVN to Git

Posted by David April 02, 2008 @ 09:39 PM

We've been preparing for Rails to move the official source repository from Subversion to [Git](#) for some time now and it seems that it'll happen over the next week or so. The premiere will happen alongside the official launch of [Github](#).

The move will also mean that we're going to be switching ticket tracking to [Lighthouse](#). So now both our repository and ticket tracking will be powered by Rails applications, which is a nice bonus treat.

When the move happens, we'll freeze the existing Subversion repository and the Trac installation. Both will live on for a long time to come, but will be entirely deprecated. This means that your existing svn:externals will not break, but if you want the latest edge, you'll have to get it from the new git repository.

So now is a great time to learn more about Git in anticipation of this move. I recommend starting with the [Git for SVN'ers crash course](#).

Perl migrates to git versioning control system.

Written on 01/03/09 at 19:14:32 by On

The [url=http://www.perlfoundation.org/]Perl Foundation[/url] has announced at the <http://use.perl.org...> website switching the version control systems to git. According to the announcement Perl 5 [url=http://use.perl.org/articles/08/12/22/0830205.shtml]migration to git[/url] would allow the language development team to take advantage of git's extensive offline and distributed versioning support.



[url=http://git.or.cz/]Git is open source[/url] and readily available to all Perl developers. Among other advantages, the announcement notes that git simplifies commits, producing fewer administrative overheads for integrating contributions. Git's change analysis tools are also singled out for praise.

The transformation from [url=http://www.perforce.com/]Perforce[/url] to git apparently took over a year. Sam Vilain of Catalyst IT "spent more than a year building custom tools to transform 21 years of Perl history into the first ever unified repository of every single change to Perl." The git repository incorporates historic snapshot releases and patch sets, which is frankly both cool and historically pleasing. Some of the patch sets were apparently recovered from old hard drives, notching up the geek satisfaction factor even more.

Git at Apache

Foundation Projects People Get Involved Support Apache Download

Git at Apache



This is a collection of read-only Git mirrors of Apache codebases. The mirrors are automatically updated and contain full version histories (including branches and tags) from the respective source trees in the official Subversion repository at Apache.

Note that these Git mirrors are missing Subversion features like svn:ignore, svn:eol-style and svn:keywords settings and support for empty directories.

Please contact the infrastructure-dev@apache.org mailing list if you have comments or suggestions regarding this service or want to see another Apache codebase mirrored.

Project	Description	Git Clone URL	HTTP Clone URL
abdera.git	Apache Abdera	git://git.apache.org/abdera.git	http://git.apache.org/abdera.git
activemq.git	Apache ActiveMQ	git://git.apache.org/activemq.git	http://git.apache.org/activemq.git
ant.git	Apache Ant	git://git.apache.org/ant.git	http://git.apache.org/ant.git
apr-iconv.git	Apache Portable Runtime iconv	git://git.apache.org/apr-iconv.git	http://git.apache.org/apr-iconv.git
apr-util.git	Apache Portable Runtime util	git://git.apache.org/apr-util.git	http://git.apache.org/apr-util.git
apr.git	Apache Portable Runtime	git://git.apache.org/apr.git	http://git.apache.org/apr.git
archiva.git	Apache Archiva	git://git.apache.org/archiva.git	http://git.apache.org/archiva.git
axis2-c.git	Apache Axis2/C	git://git.apache.org/axis2-c.git	http://git.apache.org/axis2-c.git
axis2-java.git	Apache Axis2/Java	git://git.apache.org/axis2-java.git	http://git.apache.org/axis2-java.git
batik.git	Apache Batik	git://git.apache.org/batik.git	http://git.apache.org/batik.git
buildr.git	Apache Buildr	git://git.apache.org/buildr.git	http://git.apache.org/buildr.git
camel.git	Apache Camel	git://git.apache.org/camel.git	http://git.apache.org/camel.git
cassandra.git	Apache Cassandra (incubating)	git://git.apache.org/cassandra.git	http://git.apache.org/cassandra.git
cayenne.git	Apache Cayenne	git://git.apache.org/cayenne.git	http://git.apache.org/cayenne.git
chainsaw.git	Apache Chainsaw	git://git.apache.org/chainsaw.git	http://git.apache.org/chainsaw.git
cocoon.git	Apache Cocoon	git://git.apache.org/cocoon.git	http://git.apache.org/cocoon.git
commons-codec.git	Apache Commons Codec	git://git.apache.org/commons-codec.git	http://git.apache.org/commons-codec.git
commons-collections.git	Apache Commons Collections	git://git.apache.org/commons-collections.git	http://git.apache.org/commons-collections.git

Go to "<http://git.apache.org/maven.git>"

Grails - Grails moves to Git

git grails

http://www.grails.org/blog/graeemercher/Grails+n

GRAILS > Grails moves to Git

Grails moves to Git

Blog Home

Edit Entry

Grails Moves To Git

March 25, 2009 by [graeemercher](#) | [grails git](#), [grails](#), [git](#) | [4 comments.](#)

The [Grails](#) source control repository has been officially moved to [Github](#). The [SVN repository](#) will continue to act as a read-only mirror of the Git repository.

Recent Comments:

link

A screenshot of a web browser window displaying a blog post from the Grails blog. The title of the post is "Grails Moves To Git". The post was published on March 25, 2009, by "graeemercher" and has 4 comments. The text of the post states that the Grails source control repository has been moved to Github, and the SVN repository will act as a read-only mirror. Below the post, there is a section titled "Recent Comments:" which is currently empty. The browser interface includes a toolbar at the top with various icons, a address bar with the URL "http://www.grails.org/blog/graeemercher/Grails+n", and a menu bar with "File", "Edit", "View", "Insert", "Format", "Table", "Cell", "Tools", and "Help". The overall layout is typical of a late 2000s web application.

SVN INTEROP



arguably the **best feature** of git

use git in **stealth mode**

SubVERSION

```
git svn clone http://unfurl.com/trunk  
//Hack, hack, hack  
git commit -a -m‘Some changes’  
git commit -a -m‘More changes’  
git commit -a -m‘Final changes’  
git svn dcommit
```

Session

Ambient Code 1

```
$git svn dcommit
Committing to http://ambientideas.unfuddle.com/svn/ambientideas_demo ..
.
A      anewfile2.txt
Committed r7
A      anewfile2.txt
r7 = 7772d0025962d976bcec34151155244c5af91153 (git-svn)
No changes between current HEAD and refs/remotes/git-svn
Resetting to the latest refs/remotes/git-svn
$
```

Subversion

- ▶ First class compatibility.
- ▶ Round trip support.
- ▶ Git commits = svn commits.

**effortlessly convert full history
when ready to migrate the team**

Migration

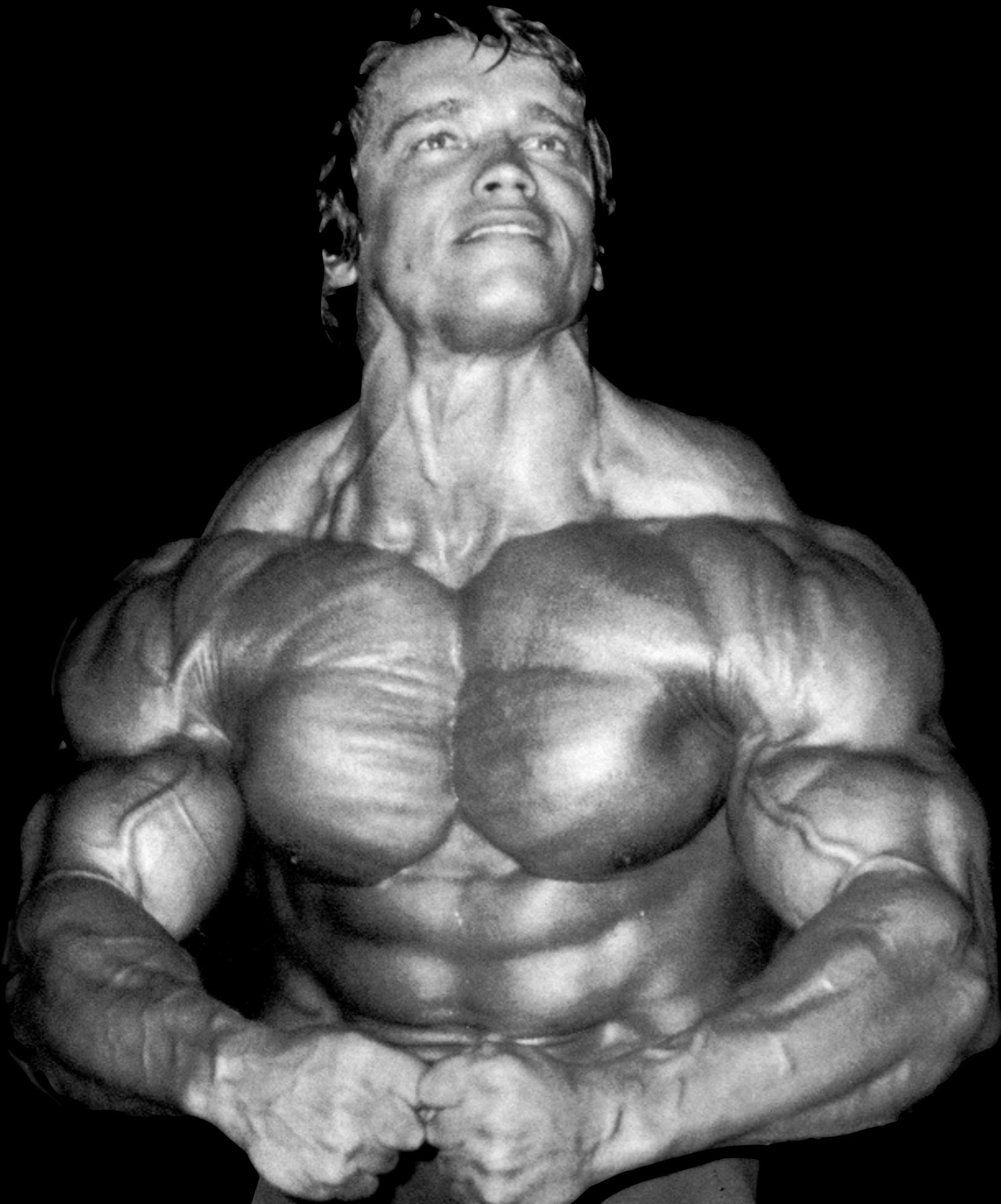
```
git svn clone http://unfurl.com/trunk
```

Migration

You now have a **full git clone** of the SVN repo

Migration

Power off your Subversion server



A black and white photograph of a muscular man from the waist up. He is shirtless, showing a well-defined abdominal muscles and large, bulging biceps. His head is tilted back, looking upwards with a focused expression. The lighting highlights the contours of his muscles against a dark background.

3
reasons

CHERRY Pick

- ▶ `git cherry-pick a5b2ee`
- ▶ Merge in just one commit.

CHERRY Pick

- ▶ `git cherry-pick a5b2ee`
- ▶ Merge in just one commit.

SEARCH HISTORY

- ▶ `git log -S "SomeText"`
- ▶ Search blob contents history without checkouts.

SEARCH HISTORY

- ▶ `git log -S "SomeText"`
- ▶ Search blob contents history without checkouts.

BISECT BUGS

- ▶ `git bisect run mvn test`
- ▶ Binary-search for bug via failed tests.
- ▶ Manual or automated modes.
- ▶ Ant, Maven, or shell script that can return 0 or 1-127

BISECT BUGS

- ▶ `git bisect run mvn test`
- ▶ Binary-search for bug via failed tests.
- ▶ Manual or automated modes.
- ▶ Ant, Maven, or shell script that can return 0 or 1-127



Twitter

@matthewmccull

Blog

<http://www.ambientideas.com/blog>

sidebar has all my social media links

Email

matthewm@ambientideas.com

GitHub

<http://github.com/matthewmccullough>

IMAGE CREDITS

- ▶ <http://www.ambientideasphotography.com>
- ▶ <http://flickr.com/photos/lenore-m/2903856664/>
- ▶ [http://en.wikipedia.org/wiki/Git_\(software\)](http://en.wikipedia.org/wiki/Git_(software))
- ▶ <http://flickr.com/photos/karenhorton/1583513014/>
- ▶ <http://flickr.com/photos/mashdnart/2545782407/>
- ▶ [http://commons.wikimedia.org/wiki/
File:Small_Boy_nuclear_test_1962.jpg](http://commons.wikimedia.org/wiki/File:Small_Boy_nuclear_test_1962.jpg)
- ▶ <http://www.flickr.com/photos/knmurphy/2506896257/>
- ▶ <http://www.flickr.com/photos/albyspace/1022035568/>
- ▶ <http://flickr.com/photos/michaelhays/3070238360/>
- ▶ http://flickr.com/photos/d_vdm/509996632/