

# Git going with DVCS

*50 minutes of mental spoon bending  
via source code control*

Hello  
my name is

*Matthew  
McCullough*





## Denver Dev Blog

Matthew McCullough's insights on software development as co-founder of Ambient Ideas, LLC



### Git at the Atlanta JUG

August 17th, 2010

Today, I'm excited to be presenting Git (my current favorite topic) to the [Atlanta JUG \(AJUG\)](#) on behalf of the [No Fluff Just Stuff Symposium Series](#). [Gunnar Hillert](#) has been most welcoming, and [Pratik Patel](#) has been a great promoter of the talk. Thank you both.

In about 75 minutes, I'll explain why the [Git Version Control System](#) deserves your attention as your next version control system. I'll show you its blazing speed adding 5000 files to a repo, creating a repository at [GitHub](#), initiating a local branch, merging with a colleague's repository, and finding which commit broke the integration tests.

I've also set up a few resources for attendees to peruse after the talk, including:

- » [Git Instructions executed during the lecture/workshop](#)

- » [Git Delicious Bookmarks](#)

- » [Git 101 Video](#)

- » [Matthew's Git DZone Article](#)

In short, if you have the least bit of dissatisfaction with your existing version control system, this talk should tip you squarely in favor of the new world of [Distributed Version Control Systems \(DVCS\)](#), and specifically, my favorite implementation, [Git](#).

Tags: [Git](#), [NFJS](#), [OpenSource](#), [Programming](#)  
Posted in [Programming](#) | [Edit](#) | [No Comments](#) »

Search for:

#### Pages

- » [About](#)

#### Archives

- » [August 2010](#)

- » [July 2010](#)

- » [May 2010](#)

- » [April 2010](#)

- » [January 2010](#)

- » [December 2009](#)

- » [October 2009](#)

- » [August 2009](#)

- » [July 2009](#)

- » [June 2009](#)

- » [May 2009](#)

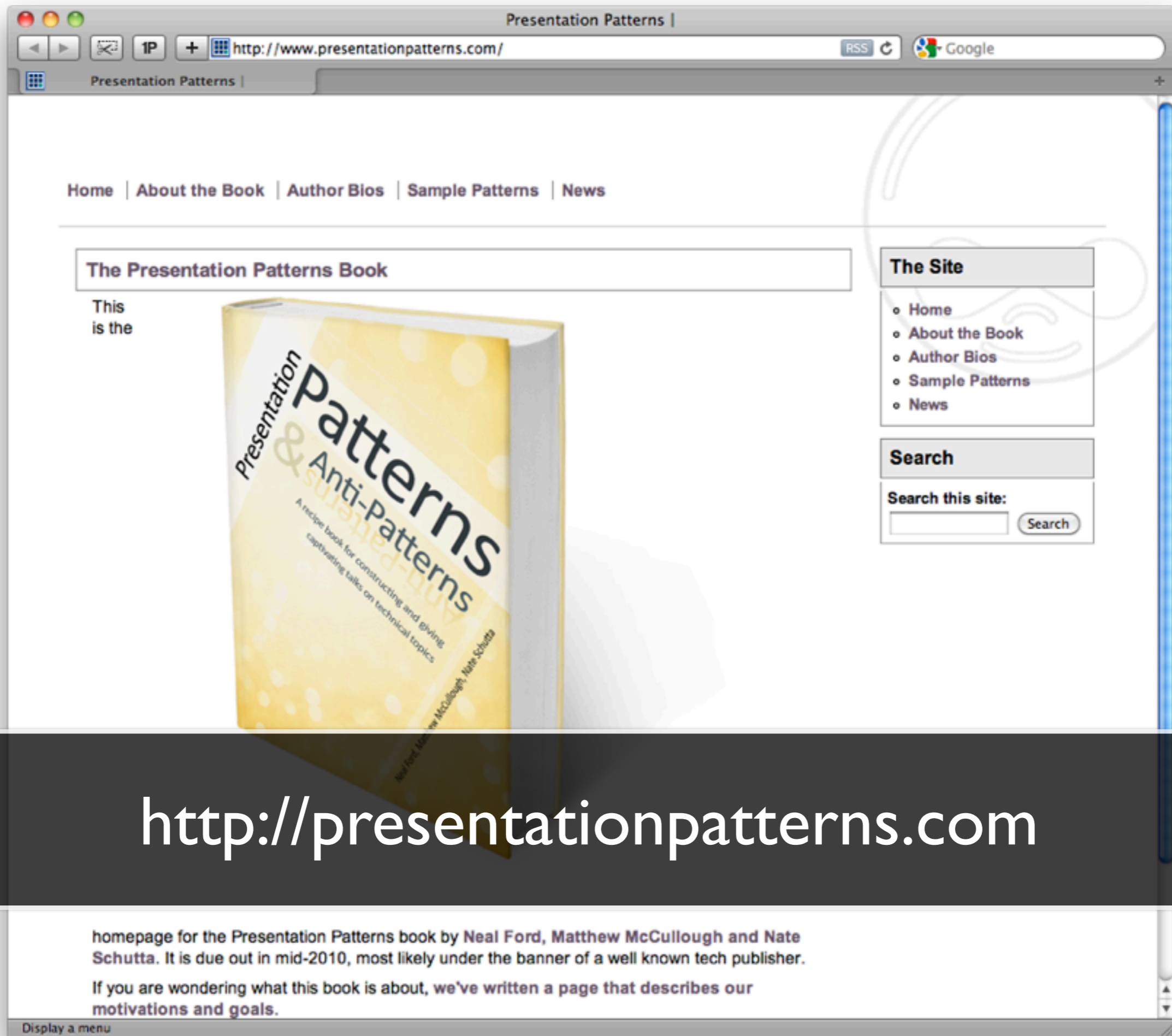
- » [April 2009](#)

- » [March 2009](#)

- » [February 2009](#)

- » [January 2009](#)

<http://ambientideas.com/blog>



<http://presentationpatterns.com>

homepage for the Presentation Patterns book by Neal Ford, Matthew McCullough and Nate Schutta. It is due out in mid-2010, most likely under the banner of a well known tech publisher.

If you are wondering what this book is about, we've written a page that describes our motivations and goals.



# O'REILLY Books & Videos

Learn something new today.

**Buy 2, Get the 3<sup>rd</sup> FREE** Use Discount Code **opc10** **All orders over \$29.<sup>95</sup> qualify for FREE SHIPPING within the US**



## Mastering Git

Matthew McCullough & Tim Berglund

VIDEO

[See larger cover](#)

## McCullough and Berglund on Mastering Git

By Matthew McCullough, Tim Berglund  
Publisher: O'Reilly Media  
Released: February 2011  
Run time: 5 hours 8 minutes

★★★★★ 0.0

[Tweet](#) 52

[Like](#) 22 people

[Write a Review](#)

[Description](#) | [Table of Contents](#) | [Related Content](#) | [Product Details](#) | [Customer Reviews](#)

**Buy 2 Get 3<sup>rd</sup> FREE** [Discount Code OPC10](#)

**Free Shipping** [On Orders > \\$29.95](#)

**100% Guarantee** [More Info](#)

### Buying Options

**Save a Tree - Go Digital** [What is this?](#)

**Video \$49.99**  
(Streaming, Downloadable)

[Add to Cart](#)

McCullough and Berglund on Mastering Git

- [View/Submit Errata](#)
- [Media Praise](#)

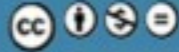
#### Got a Question?

Do you have a question about O'Reilly's products and services? Share an idea! Report a problem...

[Continue](#)

Service and support by

<http://orei.ly/ogitvid>



Wakaleo Consulting

Optimizing your software development process

**Automate your software development process**

Continuous Integration with

Hudson



<http://www.wakaleo.com/books/continuous-integration-with-hudson-the-book>

**John Ferguson Smart**



# Matthew McCullough's git Bookmarks

Bookmarks | Network | Tags | Subscriptions | Inbox

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

matthew.mccullough

git x

Type another tag

Bookmarks 118

Display options

- Save a new bookmark
- Create tag description
- Bulk edit
- Browse these bookmarks **BETA**

Tags People Options

Related Tags 77

- + agile 2
- + animation 1
- + apache 1
- + bash 1
- + blog 5
- + bugtracking 3
- + cheatsheet 10
- + clearcase 1
- + cocoa 1
- + collaboration 1
- + comics 1
- + continuousintegration 2
- + converter 2
- + development 1
- + documentation 5
- + download 1
- + dvcs 4
- + eclipse 2
- + education 3
- + free 1
- + freeware 1

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

EDIT | DELETE

opensource programming git

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.

EDIT | DELETE

macosx opensource freeware sourcecodecontrol

programming gui git

29 MAR 10 Git Client - SmartGit 246

EDIT | DELETE

sourcecodecontrol gui git

tig, the ncurses front-end to Git 158

Test based GUI for Git

EDIT | DELETE

sourcecodecontrol opensource gui git

NBGit - Git Support for NetBeans 64

EDIT | DELETE

netbeans sourcecodecontrol plugin gui opensource git

28 MAR 10 A tour of git: the basics 651

EDIT | DELETE

sourcecodecontrol howto tutorial programming git



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



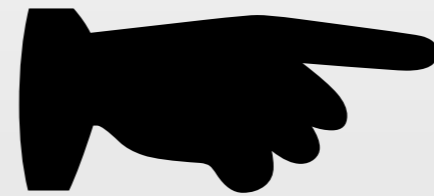


quit

**"cool kids"** version control system

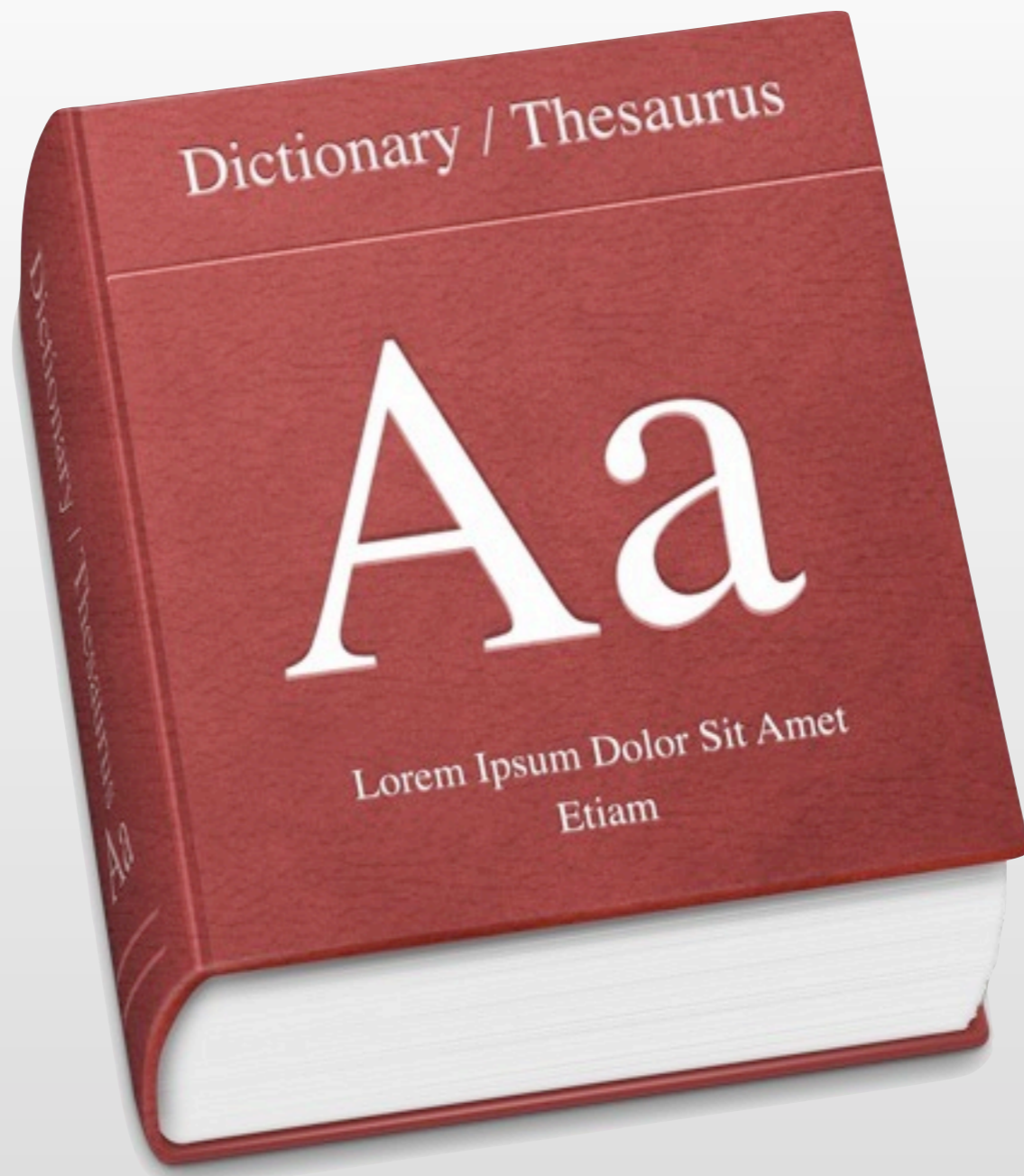
**Open Source**

**bash** scripts



**C** code





Dictionary / Thesaurus

Aa

Lorem Ipsum Dolor Sit Amet  
Etiam

“**Git**

-noun

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

-Oxford English Dictionary





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

-Linus Torvalds



Tried any DVCS?



Coming from Subversion?



Using Git today?





Certified  
Git User

**| 45+ commands**

**18** 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*

**200 I**

*arch*

**2003**

*monotone*

**2003**

*SVK*

**2003**

*darcs*

**2005**

*bazaar*

**2005**

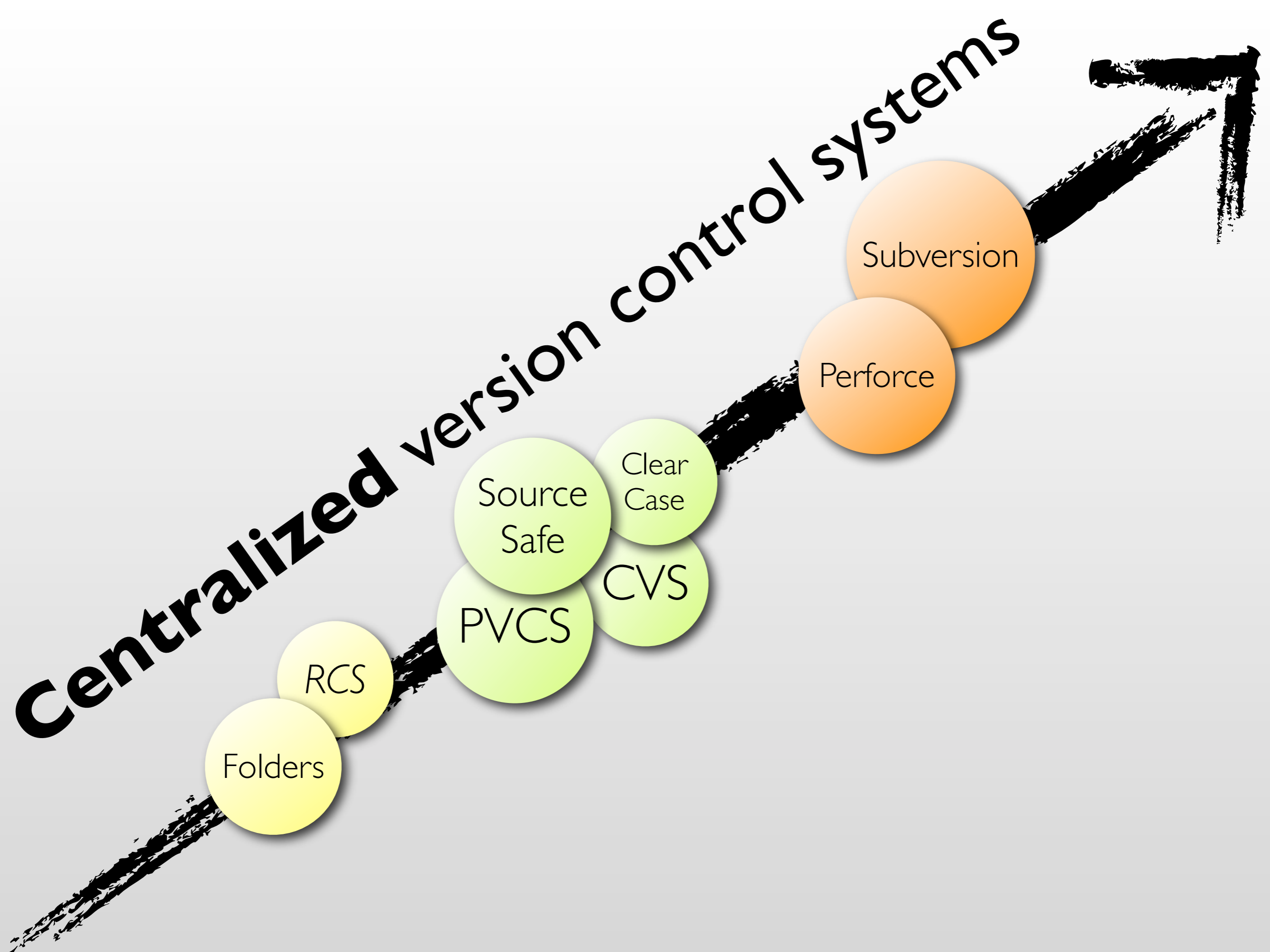
*mercurial*

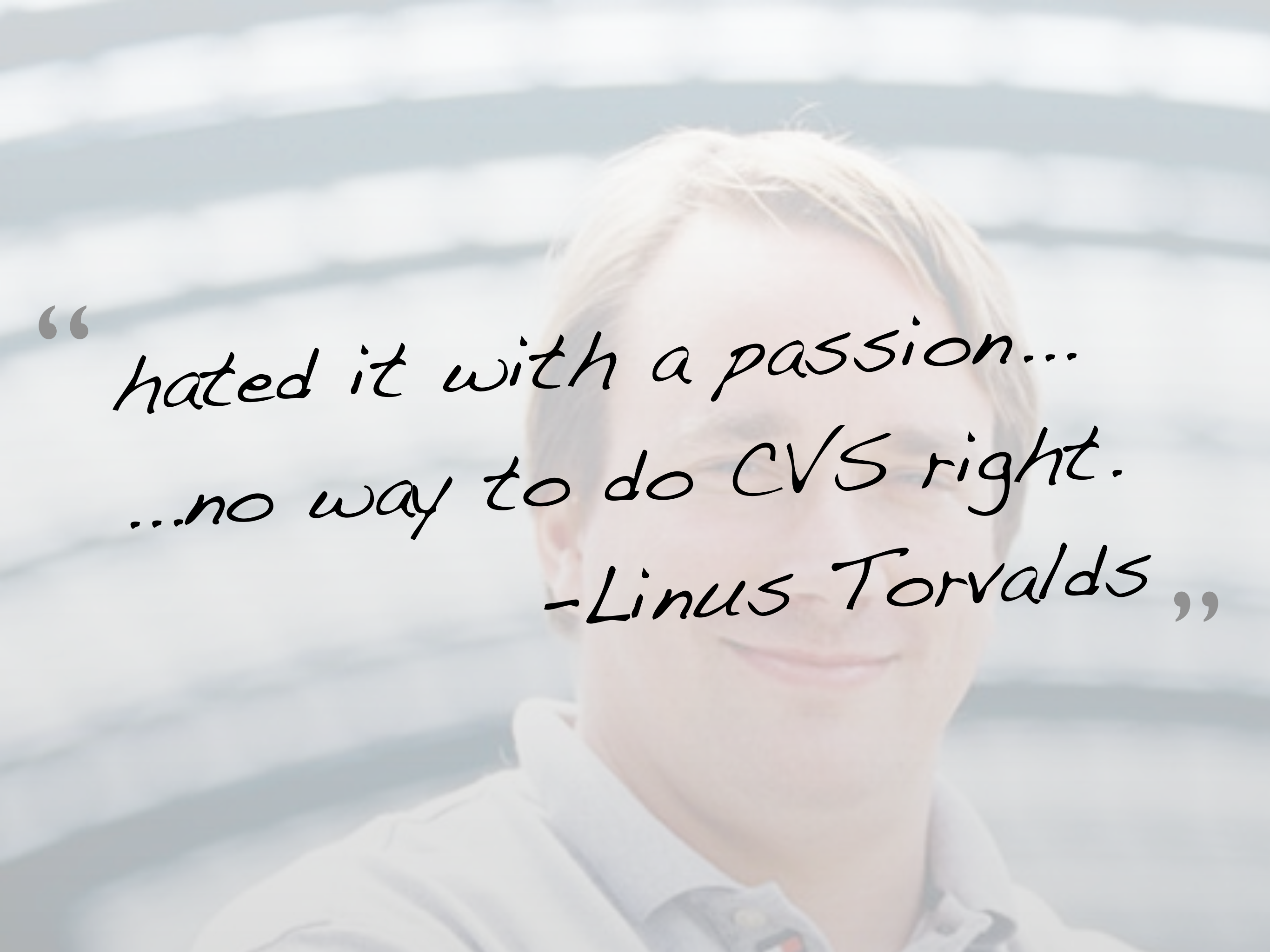


**2005**

*git*

**why?**





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

Palm Pilot

Apr 10, 97 ◀ S M T W T F S ▶

- [ 8:00 Conference Call
- 9:00
- [ 9:30 VP Operations, Q2
- 10:30
- [ 11:00 Marcom Candidate Interview
- [ 12:00 Racquetball
- [ 1:00 Lunch w/ Larry Linder
- 2:00
- [ 4:00 Staff Meeting
- 5:00
- [ 6:00 School Play

☐ New Details Go to ▾

APPLICATIONS

26°C

12.3

CALCULATOR

MENU

FIND



Why switch to

**Git**

**?**

Safely Be Wrong



Experiment



Innovate



Drive-by Assist



Crowd Source



592,048 people hosting over 1,718,058 git repositories

jQuery, reddit, Sparkle, curl, Ruby on Rails, node.js, ClickToFlash, Erlang/OTP, CakePHP, Redis, and [many more](#)



## git /'git/

Git is an extremely fast, efficient, distributed version control system ideal for the collaborative development of software.

## git·hub /'git,hʌb/

GitHub is the best way to collaborate with others. Fork, send pull requests and manage all your **public** and **private** git repositories.

**Plans, Pricing and Signup**  
Unlimited public repositories are free!

Free public repositories, collaborator management, issue tracking, wikis, downloads, code review, graphs and much more...

### Team management

**30 seconds** to give people access to code. No SSH key required. Activity feeds keep you updated on progress.

[More about collaboration](#)

### Code review

Comment on changes, track issues, compare branches, send pull requests and merge forks.

[More about code review](#)

### Reliable code hosting

We spend all day and night making sure your repositories are **secure**, **backed up** and **always available**.

[More about code hosting](#)

### Open source collaboration

Participate in the most important open source community in the world today—online or at one of our meetups.

[More about our community](#)





**and...**



**integration builds**



WHY CAN'T YOU  
EVEN KEEP THE CI  
SERVER BUILDING  
CLEANLY?

# HASHES AND Git



# INDEX vs HASH

- ▶ Centralized VCS uses DB sequential index.

# INDEX vs HASH

- ▶ Git uses SHA-1 hash.

**40 hex characters**

9AB223D28B | AA46EF | 780B22F304982E39872C34



use as **little** of it as is **unique**

9AB223D28B | AA46EF | 780B22F304982E39872C34

9AB223D28B | AA46EF | 780B22F304982E39872C34

**treeish**

treeish = **shorthand** for hashes

**9AB223**

9AB223^

**HEAD**



HEAD<sup>^</sup>

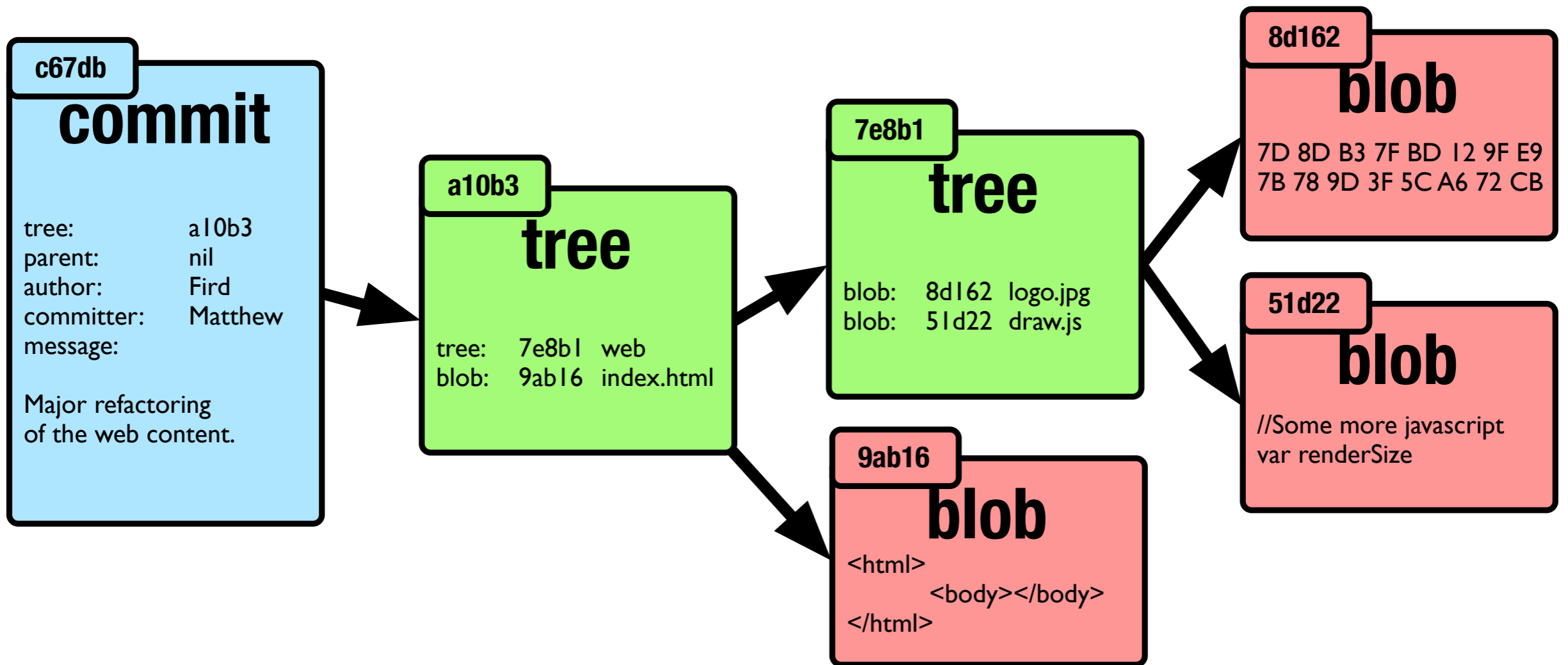
HEAD~3

`--since="2 weeks ago"`

**9AB223..56CD77**

# HASHABLE OBJECTS

- ▶ Blob
- ▶ Tree
- ▶ Commit
- ▶ Tag



Distributed  
**connectivity**



developer A



●  
developer A



●  
developer B

●  
developer A



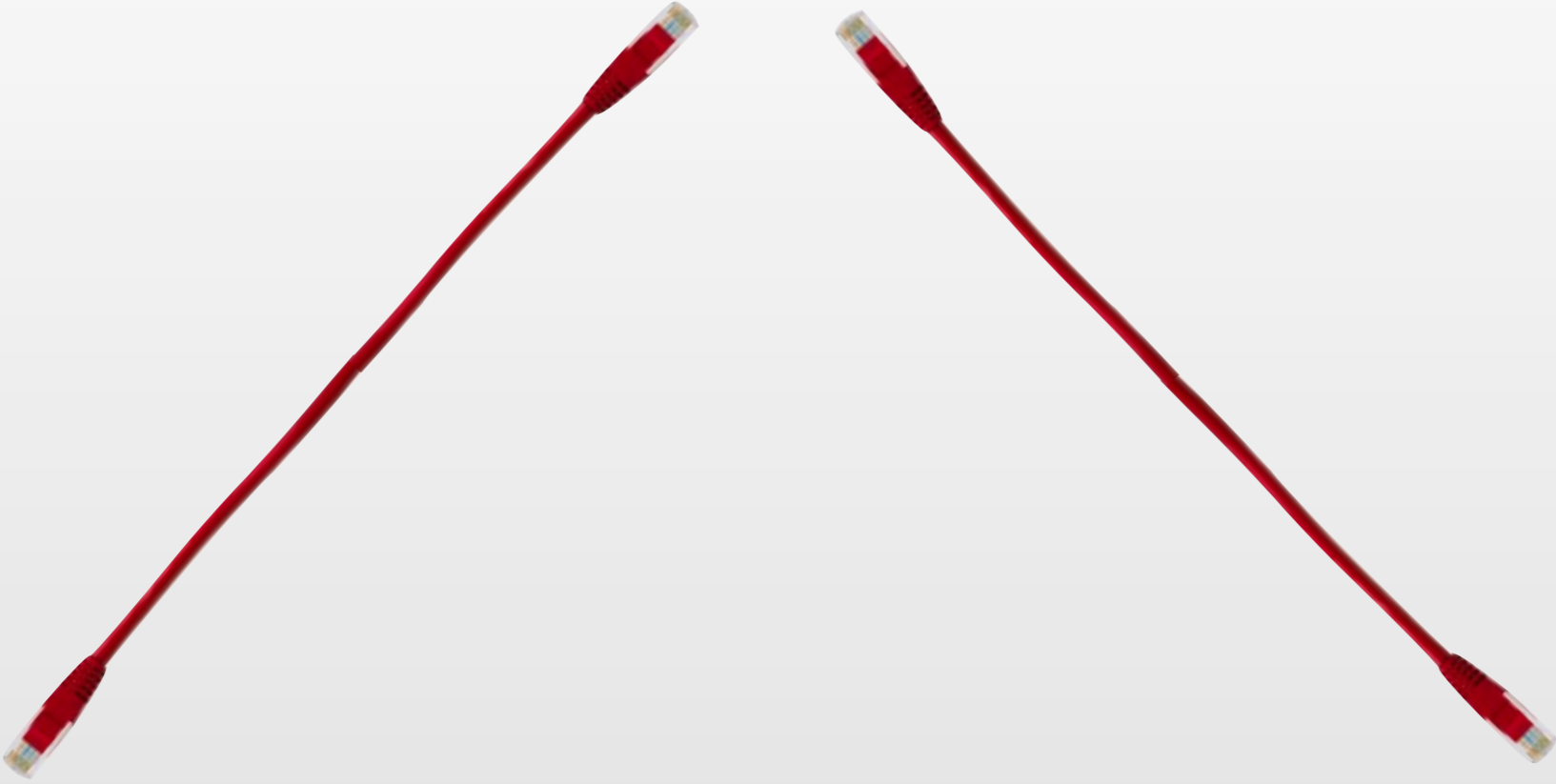
●  
developer B

●  
developer A



●  
developer B

blessed repo



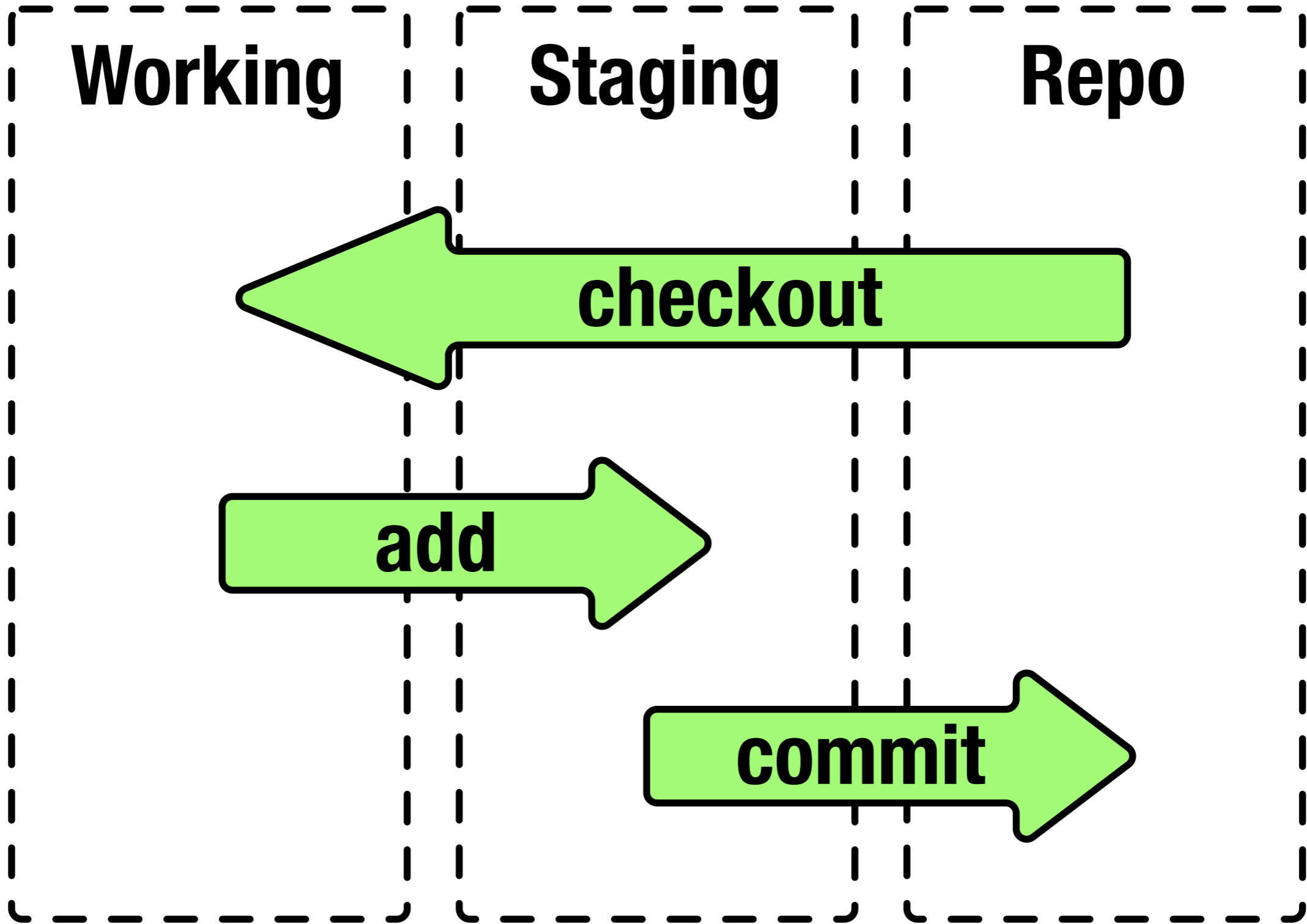
developer A



developer B



# Three Stage Thinking

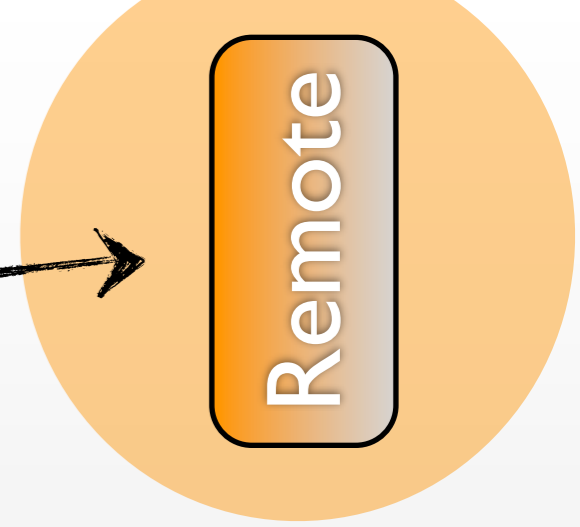




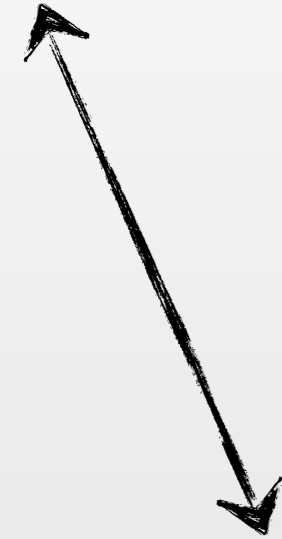
Stash



Repo



Remote



Working



Index



git clone

git checkout

edit project files

boss interrupts  
git stash

test bug on master  
working as designed

git stash apply

git add

git commit

git push

Working Offline



# More available **connectivity**





More demand to work **without**  
**connectivity**



# Offline Anything

Checkin

Add

Branch

List change log

Grep history

Rewrite history

Stash

Merge

Label

Remove

~~practically everything but push~~



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

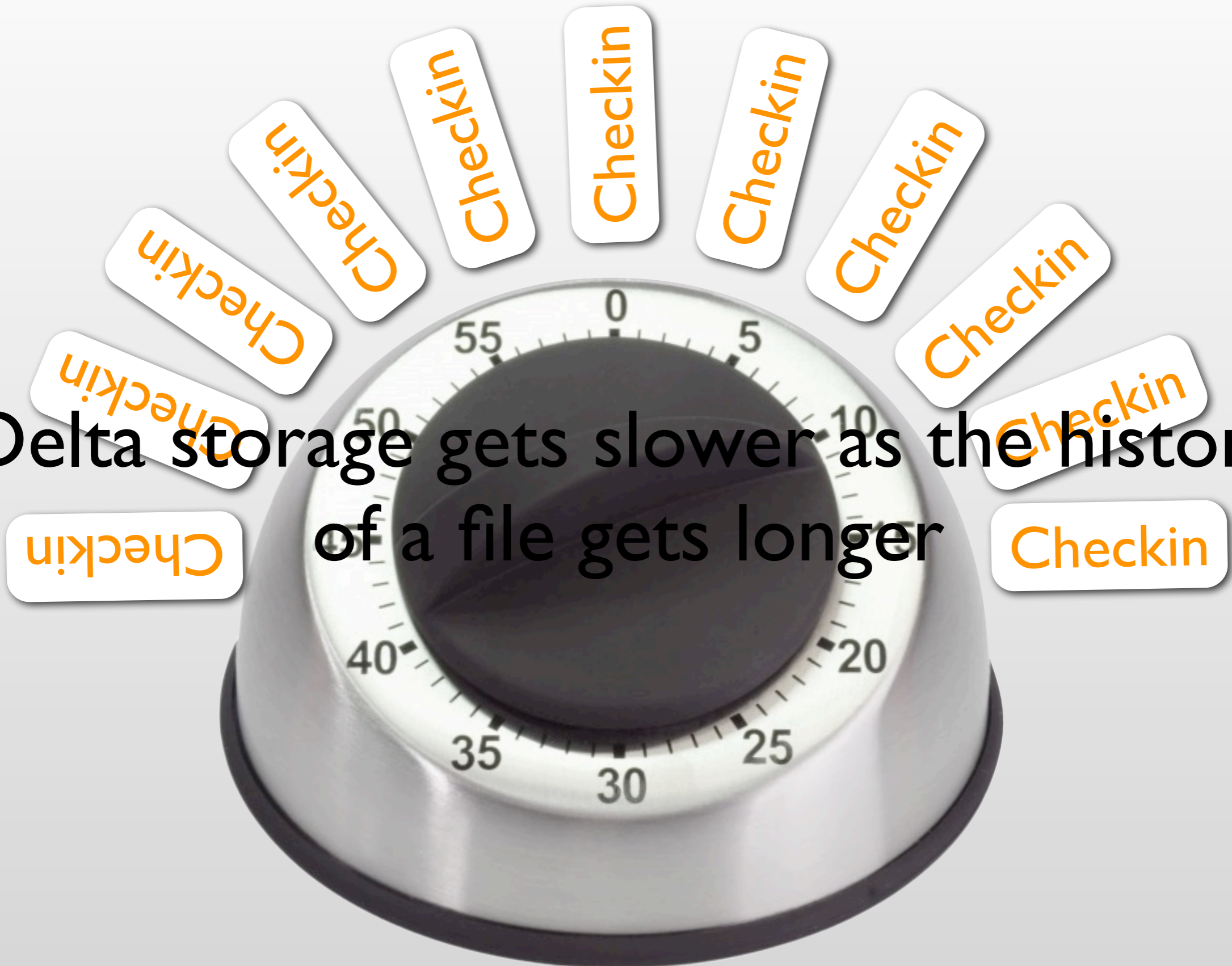
**Madelaine:** I like  
subversion  
**much better...**

# **Storage Mechanics**

Typical SCMs use **deltas**

**CVS / Subversion / darcs / Mercurial**

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





Git uses **D**irected **A**cylic **G**raph storage  
(*DAG*)

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

Copy of the **entire tree** per checkin



You've got to be kidding me!

**zlib deflates every blob at commit**

symlinks to existing **identical** blobs

**400MB** Subversion repo

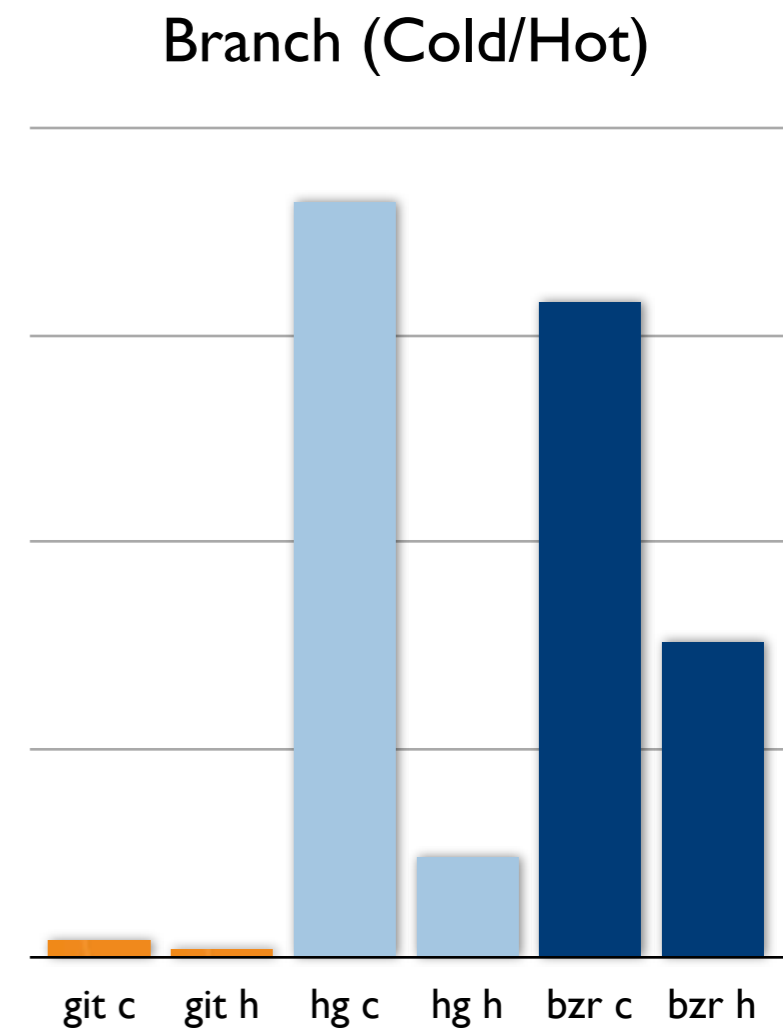
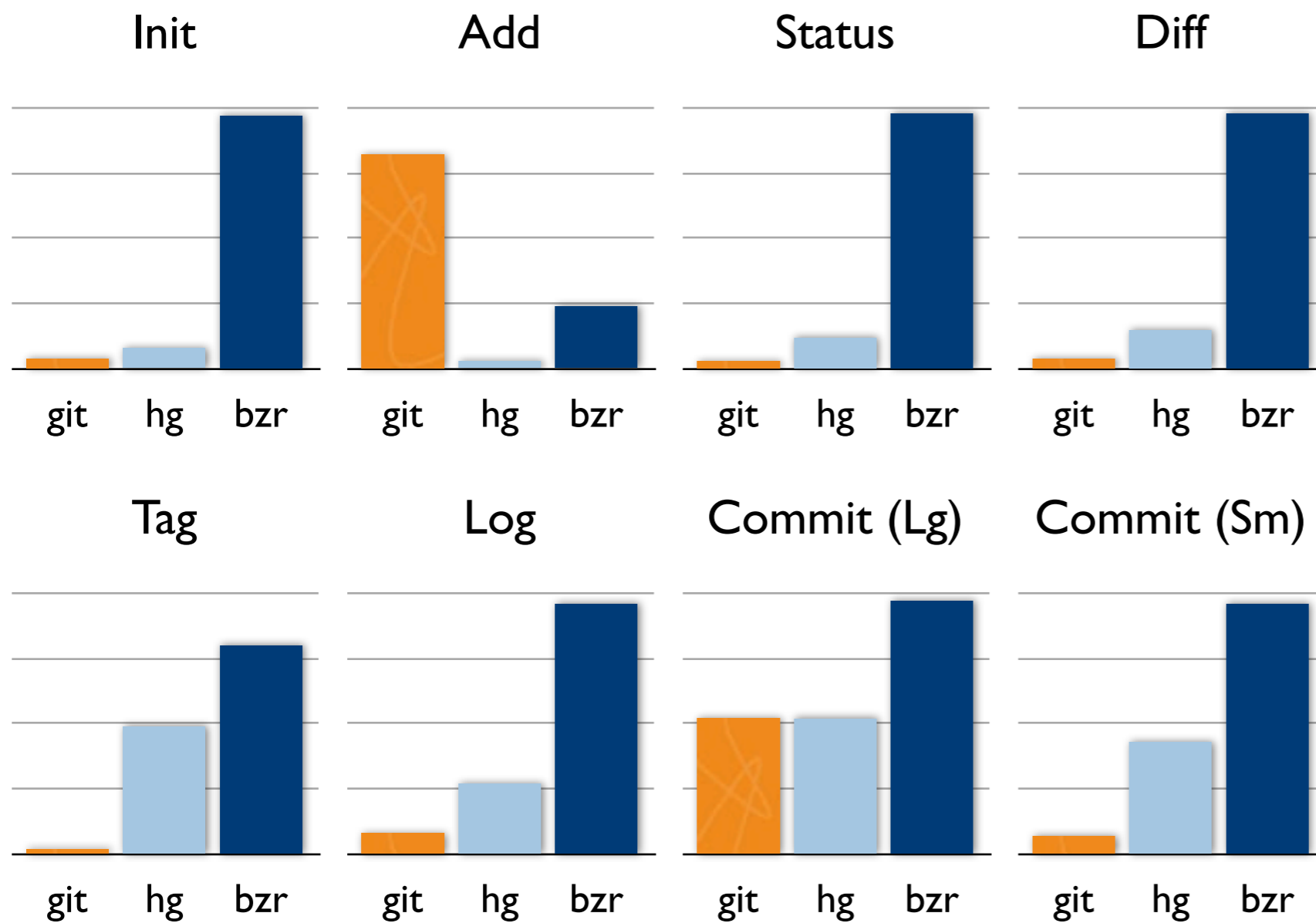
convert to a **70MB** Git repo



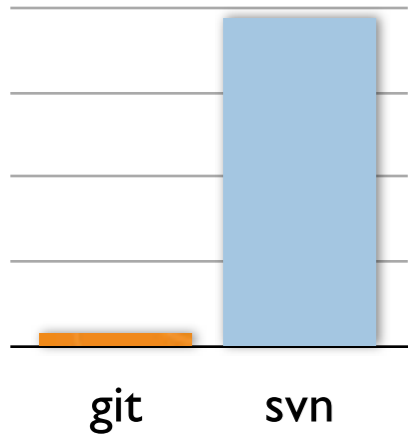


Speed

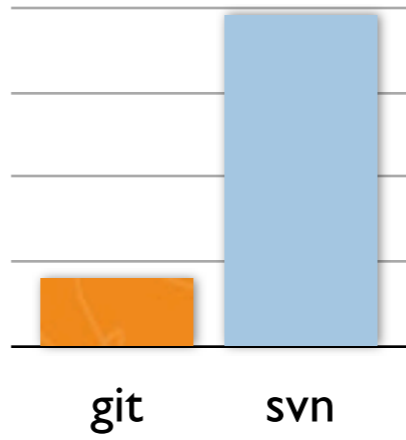
[\*\*http://whygitisbetterthanx.com\*\*](http://whygitisbetterthanx.com)



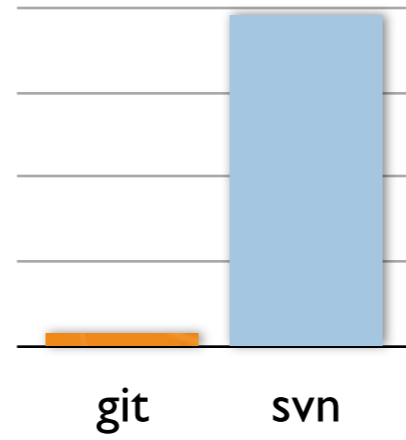
Init



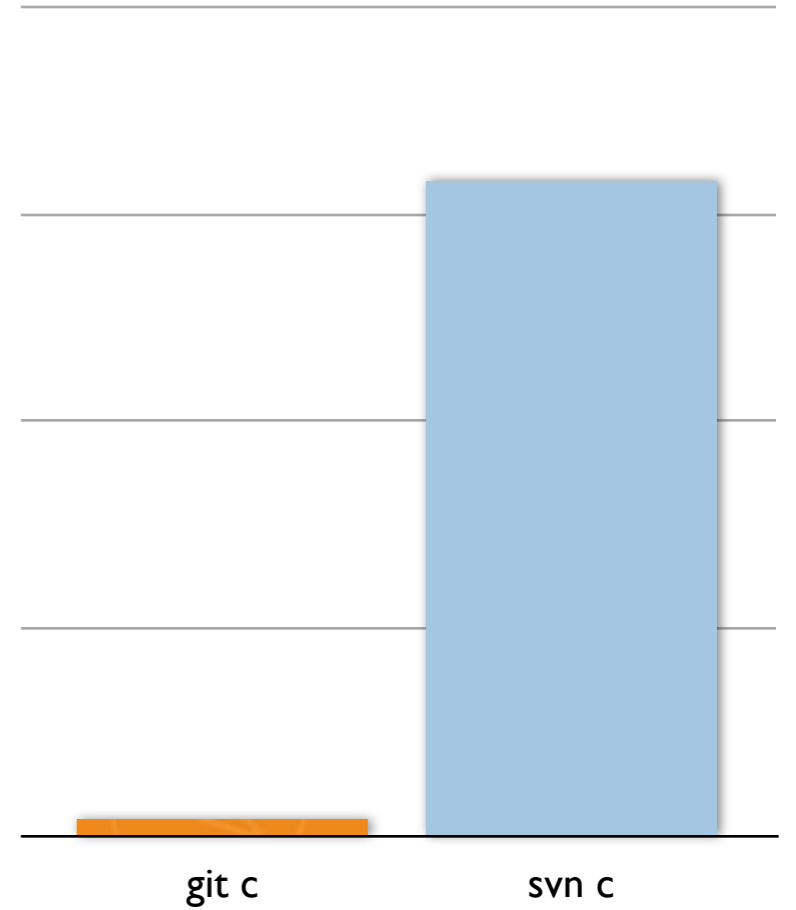
Status



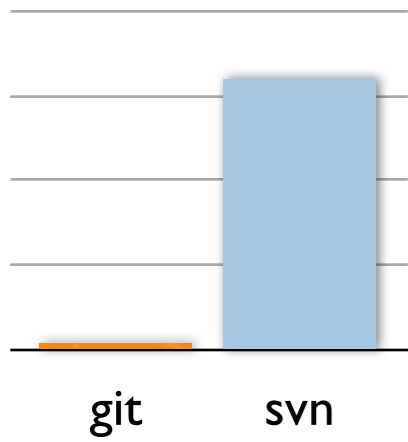
Diff



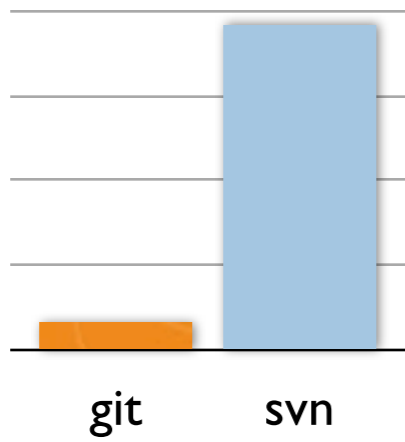
Branch



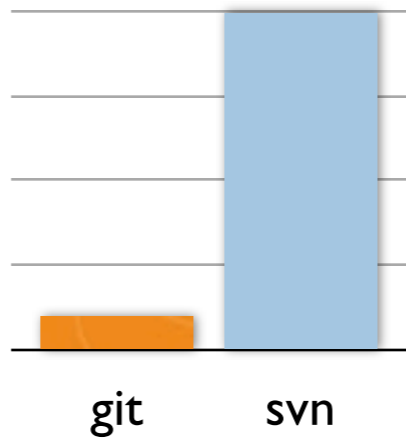
Tag



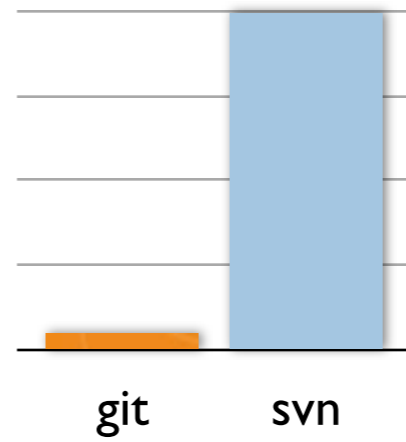
Log



Commit (Lg)



Commit (Sm)



10

TO

100 times

**FASTER**

# BRANCH, MERGE & REBASE

**Branch**

# CHEAP BRANCHES

```
git branch newbranch
```



# Rebase

Not a merge

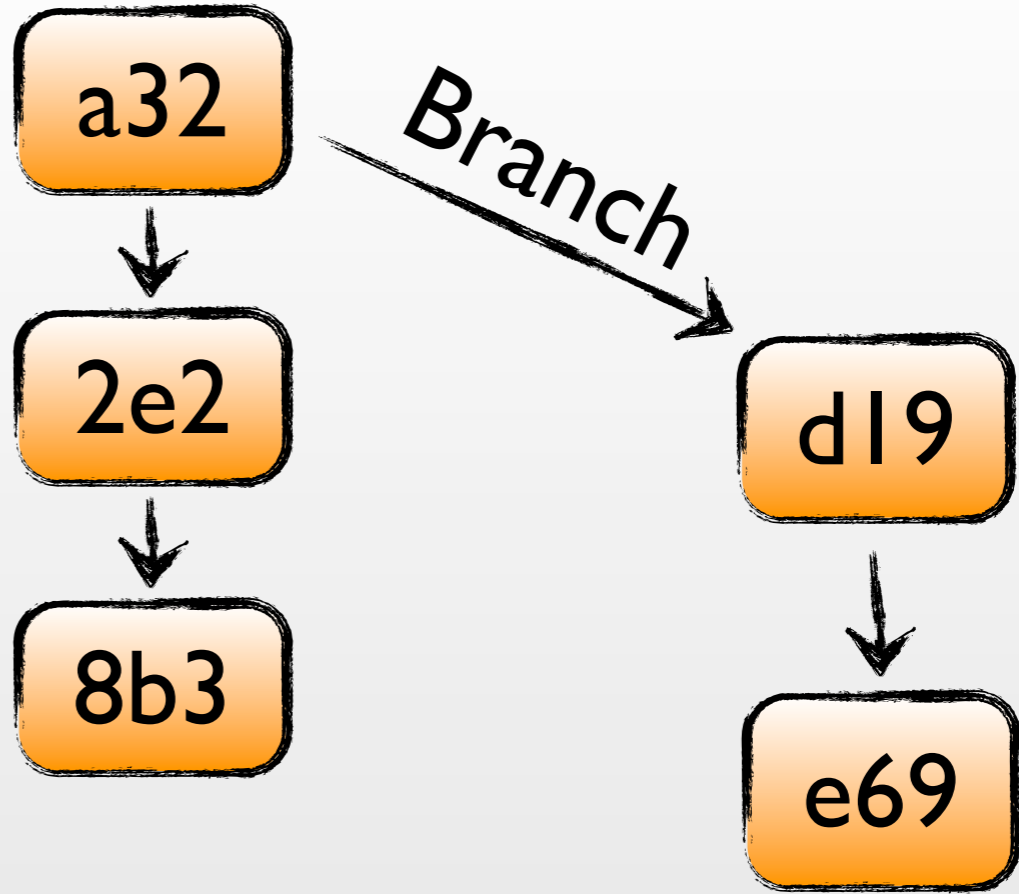
simulates team members **taking turns** working  
(one person at a time)

# REBASE for Clarity

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

# REBASING

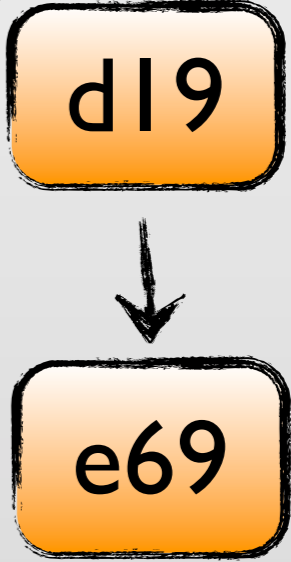
Master/Trunk/MainLatest



Master/Trunk/MainLatest

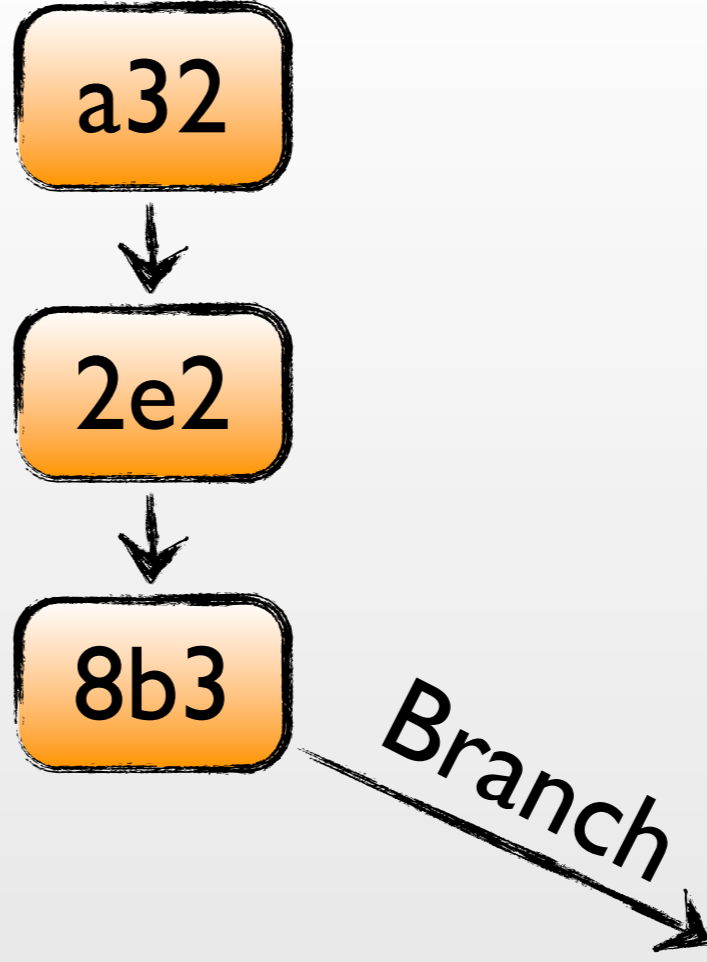


Branch

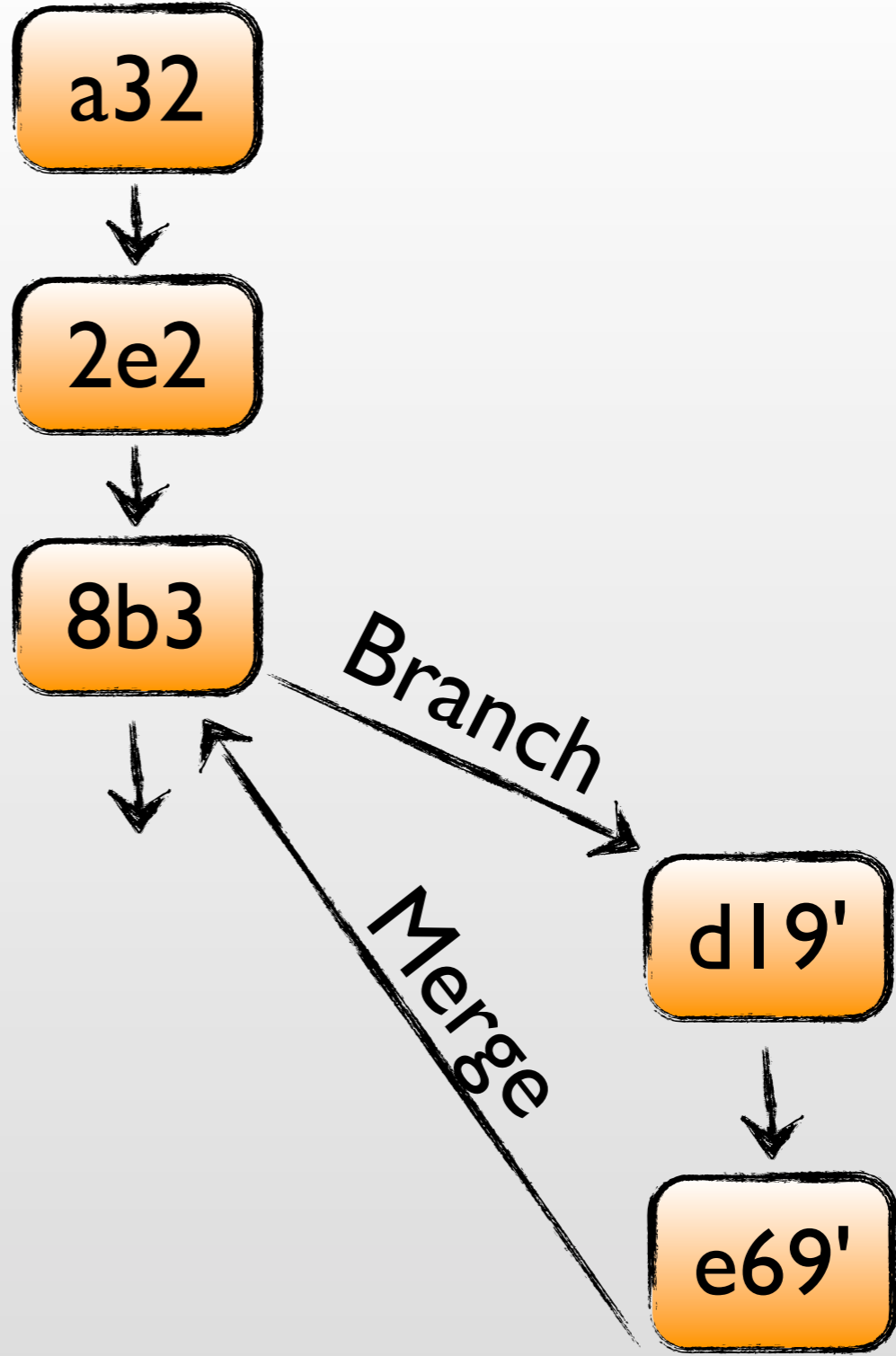


Rebase

Master/Trunk/MainLatest



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



[Git](http://git.or.cz/) is open source 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 [Perforce](http://www.perforce.com/) 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



This is an 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](mailto: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
<a href="#">abdera.git</a>	Apache Abdera	<a href="git://git.apache.org/abdera.git">git://git.apache.org/abdera.git</a>	<a href="http://git.apache.org/abdera.git">http://git.apache.org/abdera.git</a>
<a href="#">activemq.git</a>	Apache ActiveMQ	<a href="git://git.apache.org/activemq.git">git://git.apache.org/activemq.git</a>	<a href="http://git.apache.org/activemq.git">http://git.apache.org/activemq.git</a>
<a href="#">ant.git</a>	Apache Ant	<a href="git://git.apache.org/ant.git">git://git.apache.org/ant.git</a>	<a href="http://git.apache.org/ant.git">http://git.apache.org/ant.git</a>
<a href="#">apr-iconv.git</a>	Apache Portable Runtime iconv	<a href="git://git.apache.org/apr-iconv.git">git://git.apache.org/apr-iconv.git</a>	<a href="http://git.apache.org/apr-iconv.git">http://git.apache.org/apr-iconv.git</a>
<a href="#">apr-util.git</a>	Apache Portable Runtime util	<a href="git://git.apache.org/apr-util.git">git://git.apache.org/apr-util.git</a>	<a href="http://git.apache.org/apr-util.git">http://git.apache.org/apr-util.git</a>
<a href="#">apr.git</a>	Apache Portable Runtime	<a href="git://git.apache.org/apr.git">git://git.apache.org/apr.git</a>	<a href="http://git.apache.org/apr.git">http://git.apache.org/apr.git</a>
<a href="#">archiva.git</a>	Apache Archiva	<a href="git://git.apache.org/archiva.git">git://git.apache.org/archiva.git</a>	<a href="http://git.apache.org/archiva.git">http://git.apache.org/archiva.git</a>
<a href="#">axis2-c.git</a>	Apache Axis2/C	<a href="git://git.apache.org/axis2-c.git">git://git.apache.org/axis2-c.git</a>	<a href="http://git.apache.org/axis2-c.git">http://git.apache.org/axis2-c.git</a>
<a href="#">axis2-java.git</a>	Apache Axis2/Java	<a href="git://git.apache.org/axis2-java.git">git://git.apache.org/axis2-java.git</a>	<a href="http://git.apache.org/axis2-java.git">http://git.apache.org/axis2-java.git</a>
<a href="#">batik.git</a>	Apache Batik	<a href="git://git.apache.org/batik.git">git://git.apache.org/batik.git</a>	<a href="http://git.apache.org/batik.git">http://git.apache.org/batik.git</a>
<a href="#">buildr.git</a>	Apache Buildr	<a href="git://git.apache.org/buildr.git">git://git.apache.org/buildr.git</a>	<a href="http://git.apache.org/buildr.git">http://git.apache.org/buildr.git</a>
<a href="#">camel.git</a>	Apache Camel	<a href="git://git.apache.org/camel.git">git://git.apache.org/camel.git</a>	<a href="http://git.apache.org/camel.git">http://git.apache.org/camel.git</a>
<a href="#">cassandra.git</a>	Apache Cassandra (incubating)	<a href="git://git.apache.org/cassandra.git">git://git.apache.org/cassandra.git</a>	<a href="http://git.apache.org/cassandra.git">http://git.apache.org/cassandra.git</a>
<a href="#">cayenne.git</a>	Apache Cayenne	<a href="git://git.apache.org/cayenne.git">git://git.apache.org/cayenne.git</a>	<a href="http://git.apache.org/cayenne.git">http://git.apache.org/cayenne.git</a>
<a href="#">chainsaw.git</a>	Apache Chainsaw	<a href="git://git.apache.org/chainsaw.git">git://git.apache.org/chainsaw.git</a>	<a href="http://git.apache.org/chainsaw.git">http://git.apache.org/chainsaw.git</a>
<a href="#">cocoon.git</a>	Apache Cocoon	<a href="git://git.apache.org/cocoon.git">git://git.apache.org/cocoon.git</a>	<a href="http://git.apache.org/cocoon.git">http://git.apache.org/cocoon.git</a>
<a href="#">commons-codec.git</a>	Apache Commons Codec	<a href="git://git.apache.org/commons-codec.git">git://git.apache.org/commons-codec.git</a>	<a href="http://git.apache.org/commons-codec.git">http://git.apache.org/commons-codec.git</a>
<a href="#">commons-collections.git</a>	Apache Commons Collections	<a href="git://git.apache.org/commons-collections.git">git://git.apache.org/commons-collections.git</a>	<a href="http://git.apache.org/commons-collections.git">http://git.apache.org/commons-collections.git</a>



# Grails moves to Git

> Grails moves to Git

- [Blog Home](#)
- [Edit Entry](#)

## Grails Moves To Git

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

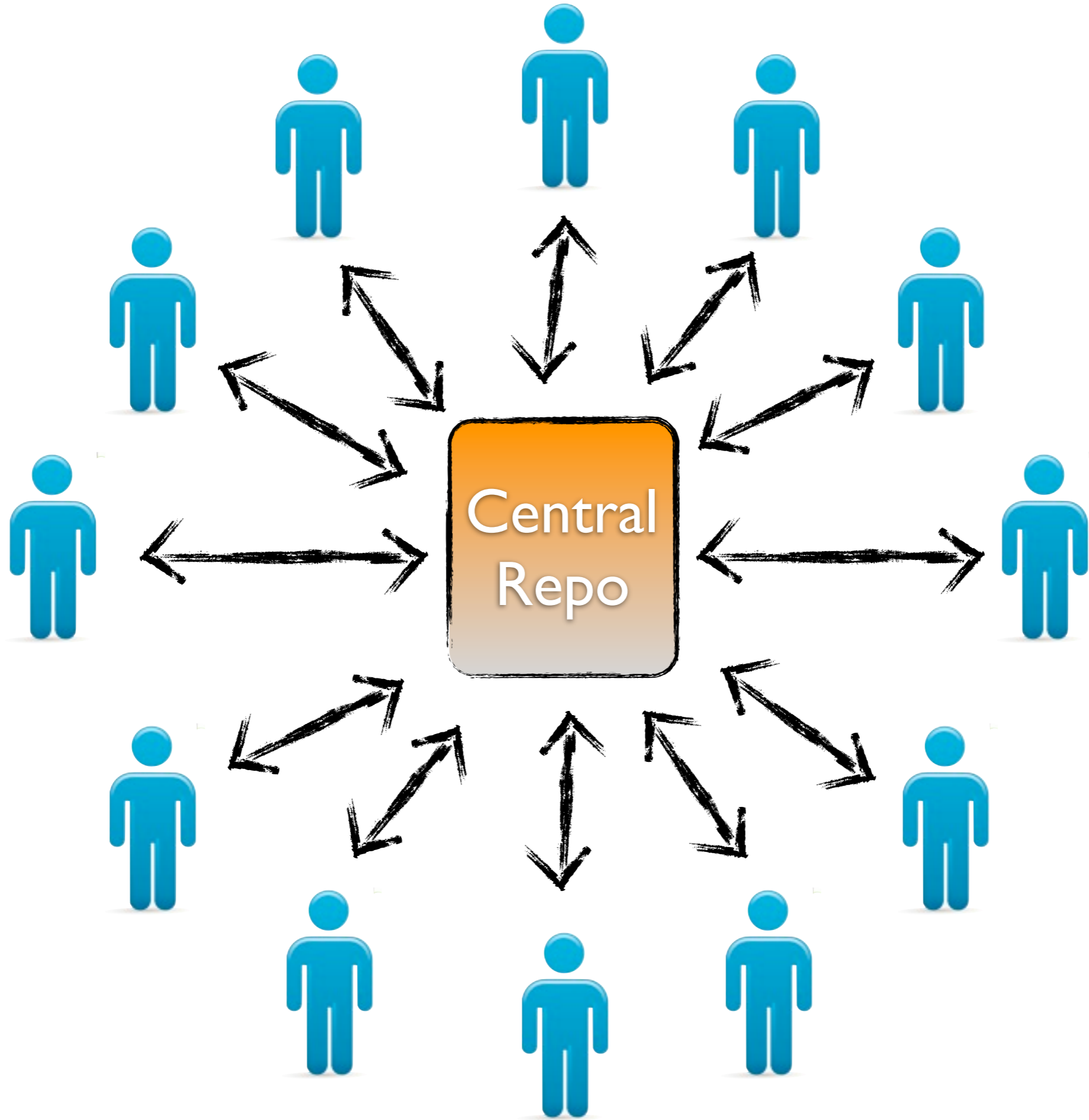
The [Grails](#) source control repository has been officially moved to [Github](#). The [SVN repository](#) will continue to act as a read-only mirror of the Git repository.

### Recent Comments:

[link](#)

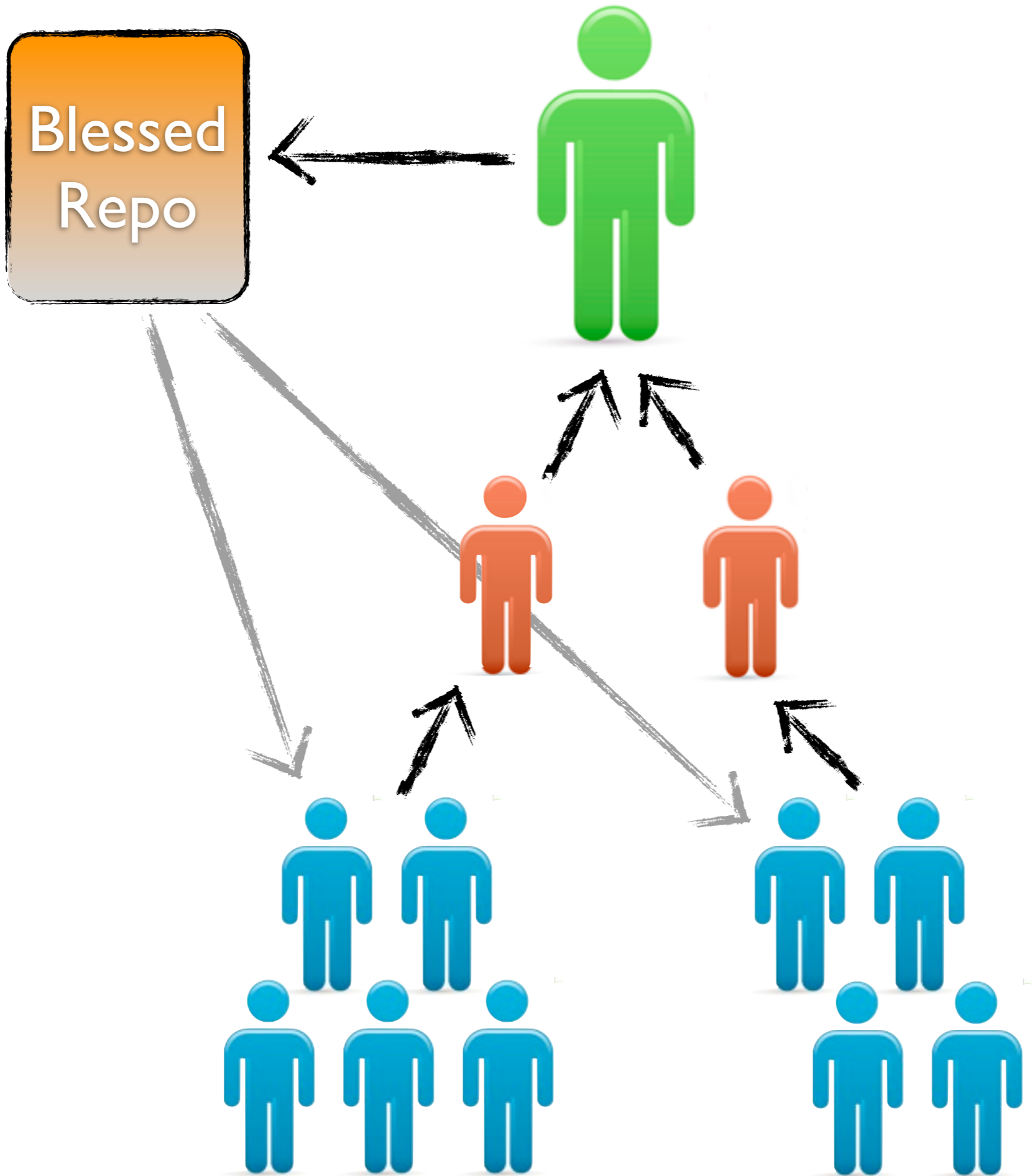
# Usage Models



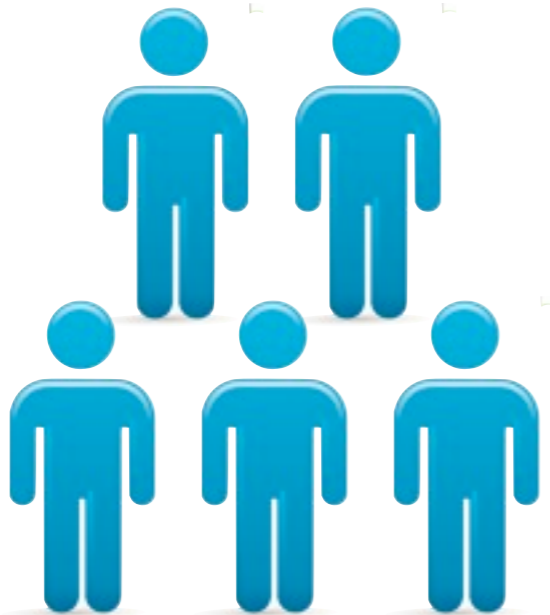
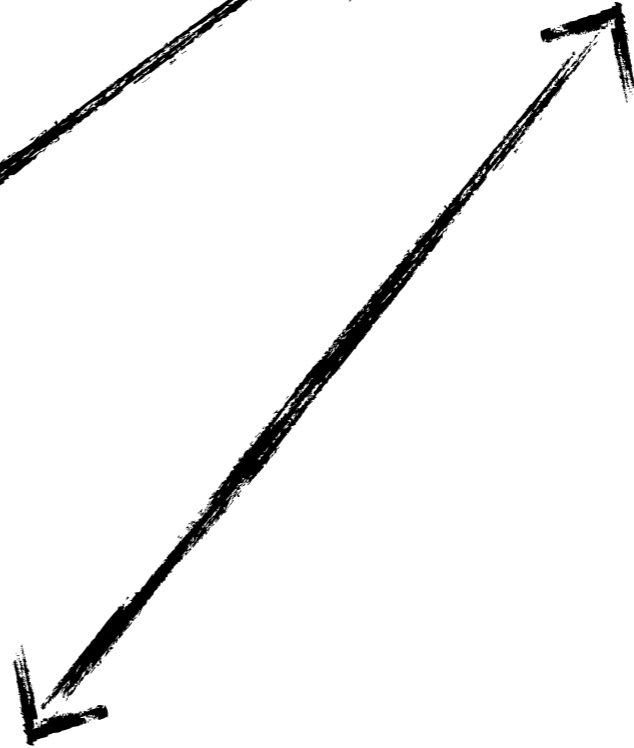
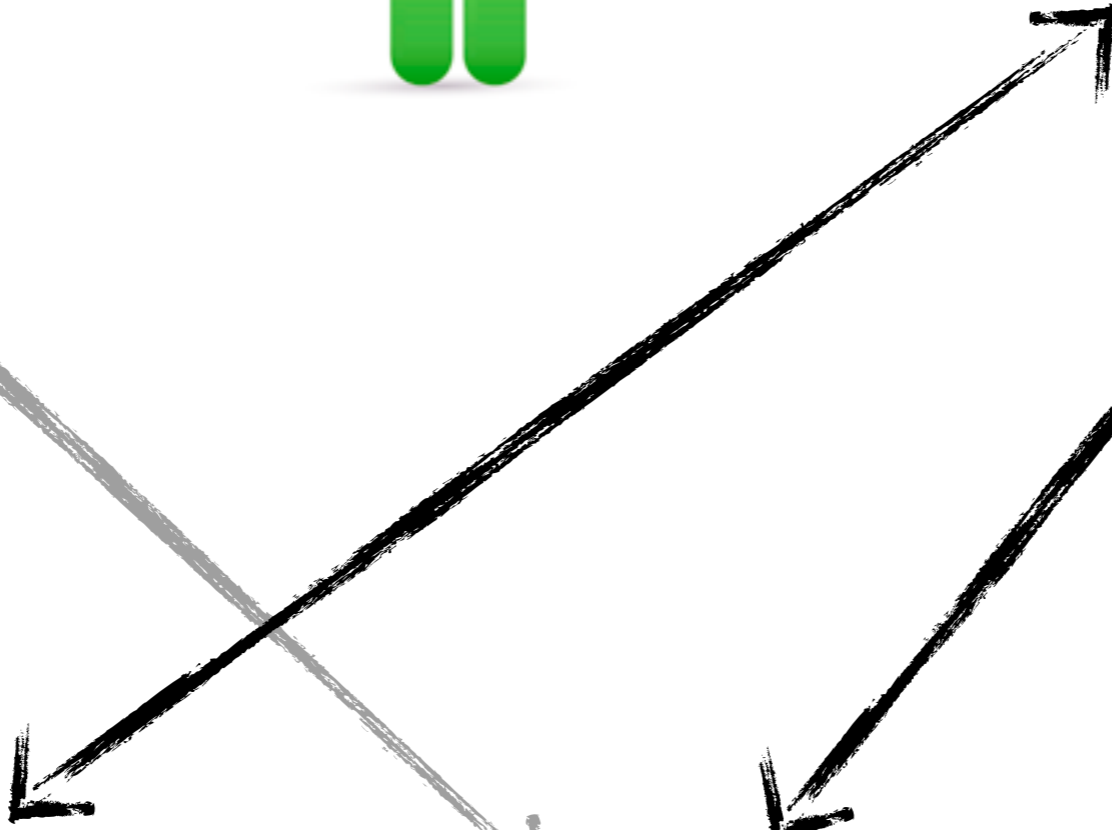
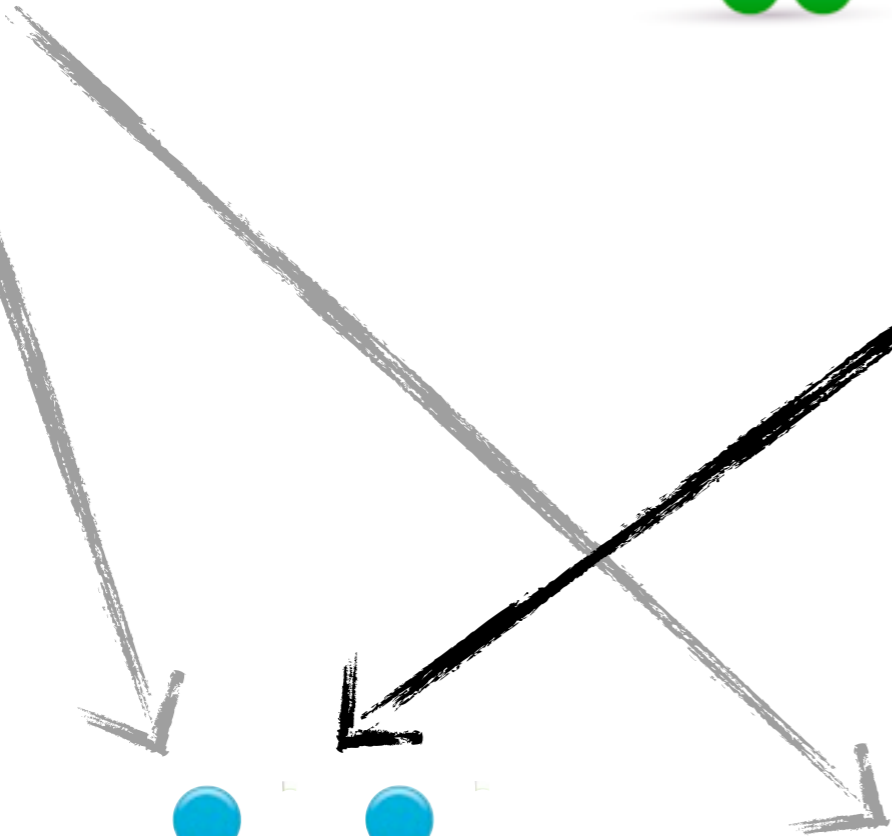
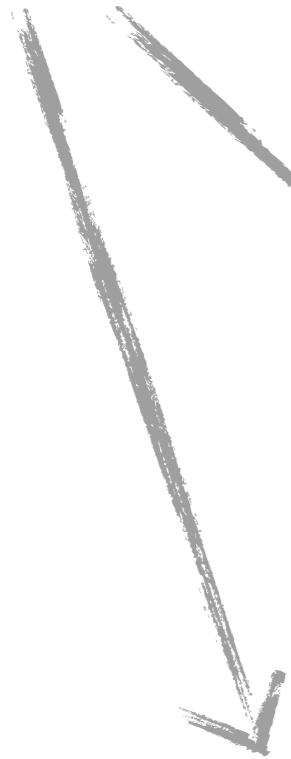


Centralized





Dictatorship



Integration Managed

# 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
```

```
$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



3  
reasons

# CHERRY Pick

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

# SEARCH HISTORY

- ▶ `git grep SomeText HEAD^^^`
- ▶ 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





Twitter

@matthewmccull

Blog

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

sidebar has all my social media links

Email

matthewm@ambientideas.com

GitHub

<http://github.com/matthewmccullough>

