



# Understanding and Using Git at Eclipse

<http://eclipse.org/egit>

Chris Aniszczyk (Red Hat)

[zx@redhat.com](mailto:zx@redhat.com)

<http://aniszczyk.org>



Shawn Pearce (Google)

[sop@google.com](mailto:sop@google.com)

<http://spearce.org>

# Agenda



- Introduction
- How Git Works
- Demos
- Conclusion
- Q&A

# Git and Eclipse



**Git** is a Distributed Version Control System (DVCS, GPL)

**EGit** is an Eclipse Team provider for Git (EPL).

**JGit** is a lightweight Java library implementing Git (EDL).

Technology sub projects (incubation)

- <http://www.eclipse.org/egit>
- <http://www.eclipse.org/jgit>

The goal is to **build a community** around Git at Eclipse

# EGit and JGit Diverse Committership




- Chris Aniszczyk (Red Hat)
- Christian Halstrick (SAP)
- Mik Kersten (Tasktop)
- Mathias Kinzler (SAP)
- Stefan Lay (SAP)
- Mykola Nikishov (Independent)
- Shawn Pearce (Google)
- Robin Rosenberg (Dewire)
- Matthias Sohn (SAP)
- Remy Suen (IBM)
- Gunnar Wagenknecht (Ageto)

# Git at Eclipse



Eclipse is moving to Git...

<http://git.eclipse.org>



## Eclipse.org Git repositories

To use Git in Eclipse, check out the EGit project.

[Index](#) [about](#)

This is a web interface for Eclipse Git repositories. You can also [browse our CVS and SVN repositories](#). For more information about Git, please see the [Git Wiki page](#).

The contents of the Git repositories are made available under the terms and conditions of the [Eclipse.org Software User Agreement](#).

Name	Description	Owner	Idle
bpmn2	Unnamed repository; edit this file 'description' to name the repository.	Antoine Toulme	20 hours
bpmnmodeler	Unnamed repository; edit this file 'description' to name the repository.	Antoine Toulme	10 days
emf/org.eclipse.emf.mwe.git	Unnamed repository; edit this file 'description' to name the repository.	root	
gemini.web/org.eclipse.gemini.web.gemini-web-container.git	Gemini Web - OSGi web container implementation	Glyn Normington	13 hours
gyrex/main.git	Gyrex - Equinox on servers.	Gunnar Wagenknecht	8 weeks
mpc/org.eclipse.epp.mpc.git	Unnamed repository; edit this file 'description' to name the repository.	Steffen Pingel	5 weeks
mylyn/org.eclipse.mylyn.reviews.git	Unnamed repository; edit this file 'description' to name the repository.	Mario Bernhart	4 days
scalamodules/org.eclipse.scalamodules.git	Unnamed repository; edit this file 'description' to name the repository.	Heiko Seeberger	3 months
scalamodules/org.eclipse.scalamodules.reponame.git	Unnamed repository; edit this file 'description' to name the repository.	Heiko Seeberger	
sketch/org.eclipse.sketch.git	Eclipse Sketch project	Ugo Braga Sangiorgi	8 weeks
swtbot/org.eclipse.swtbot.git	Unnamed repository; edit this file 'description' to name the repository.	Ketan Padegaonkar	5 months
tmt/org.eclipse.xtext.git	Unnamed repository; edit this file 'description' to name the repository.	root	11 hours
tmt/org.eclipse.xtext.releng.git	Unnamed repository; edit this file 'description' to name the repository.	root	11 hours
virgo/org.eclipse.virgo.apps.git	Virgo Applications - Virgo-supplied applications	Glyn Normington	6 days

# History



- 2005     Linus Torvalds starts Git
- 2006     Proof-of-concept, quite unusable
- 2007     Index reader, quickdiff
- 2008     Add history view, commit, push/fetch
- 2009     Eclipse decides on Git; moved to Eclipse.org
  
- 3/2010   Released 0.7  
          Diff/Merge Algorithms, Automatic IP Logs
- 6/2010   Released 0.8 (Helios)  
          Usability Improvements, Git Repositories View, Tagging
  
- 9/2010   Planned 0.9 (Helios SR1)  
          Merge, Synchronize View, Stashing, Staging
  
- 12/2010   Planned 1.0

# Git, Eclipse and Rome



EGit is still beta (incubation by Eclipse terms) and we want to establish a feedback loop to improve the tooling

“Rome wasn’t built in a day”





# Features in EGit 0.8

- |                        |                    |
|------------------------|--------------------|
| ✓ git init / git clone | ✗ git pull         |
| ✓ git add              | ✓ git push         |
| ✓ git status           | ✗ git stash*       |
| ✓ git commit           | ✓ git branch       |
| ✓ git diff             | ✓ git tag          |
| ✓ git fetch            | ✓ git checkout     |
| ✓ git log              | ✓ git config*      |
| ✓ git merge*           | ✓ git format-patch |
| ✗ git rebase           | ✓ git mv* / git rm |
| ✓ git remote           | ✓ git reset        |

- ✓ Supported
- ✓ Partially supported
- ✗ Not yet supported  
\* planned for 0.9  
in September '10



# No Free Lunch



The best way to learn Git is to use Git





# Git vs. CVS/SVN

- **Distributed (git)**

- Full local history
- Work off-line
- Fast
- Rebase patches easily
- Forks happen, deal with it
- Powerful merging, allows many branches

- **Centralized (CVS)**

- No
- No
- Slow
- Patches go stale
- Forks are painful
- Merging is painful

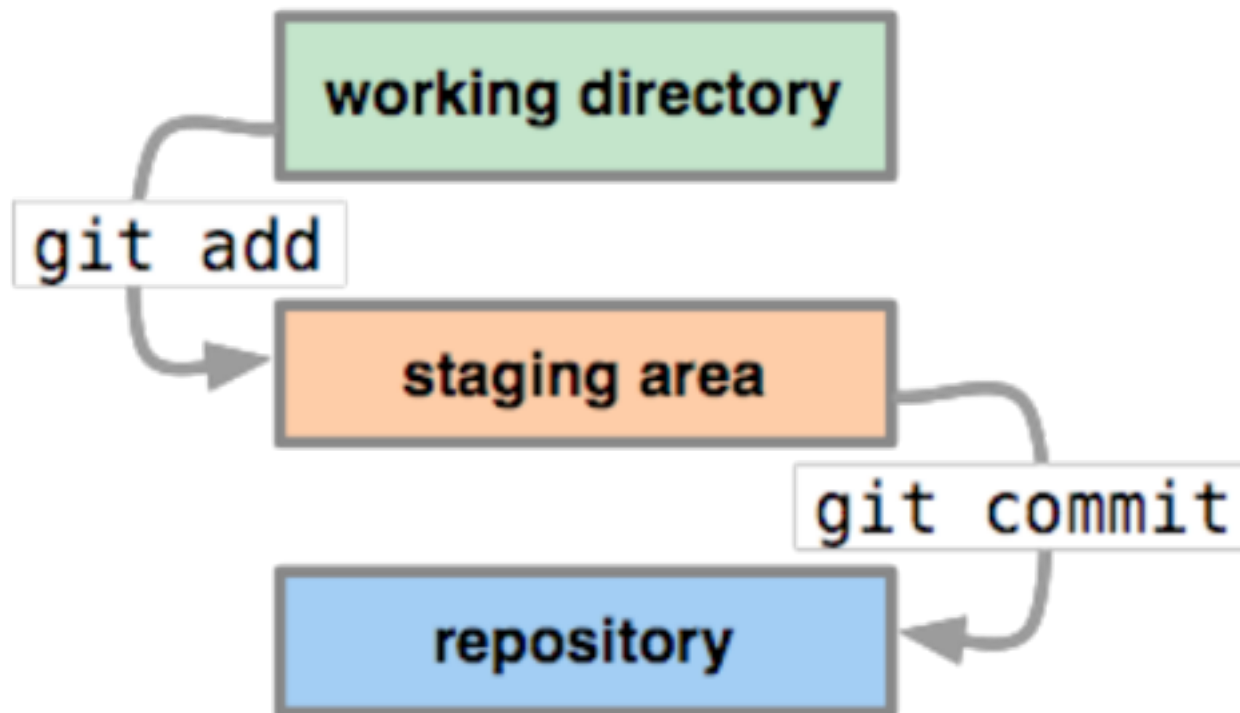
# Agenda



- Introduction
- **How Git Works**
- Demos
- Conclusion
- Q&A

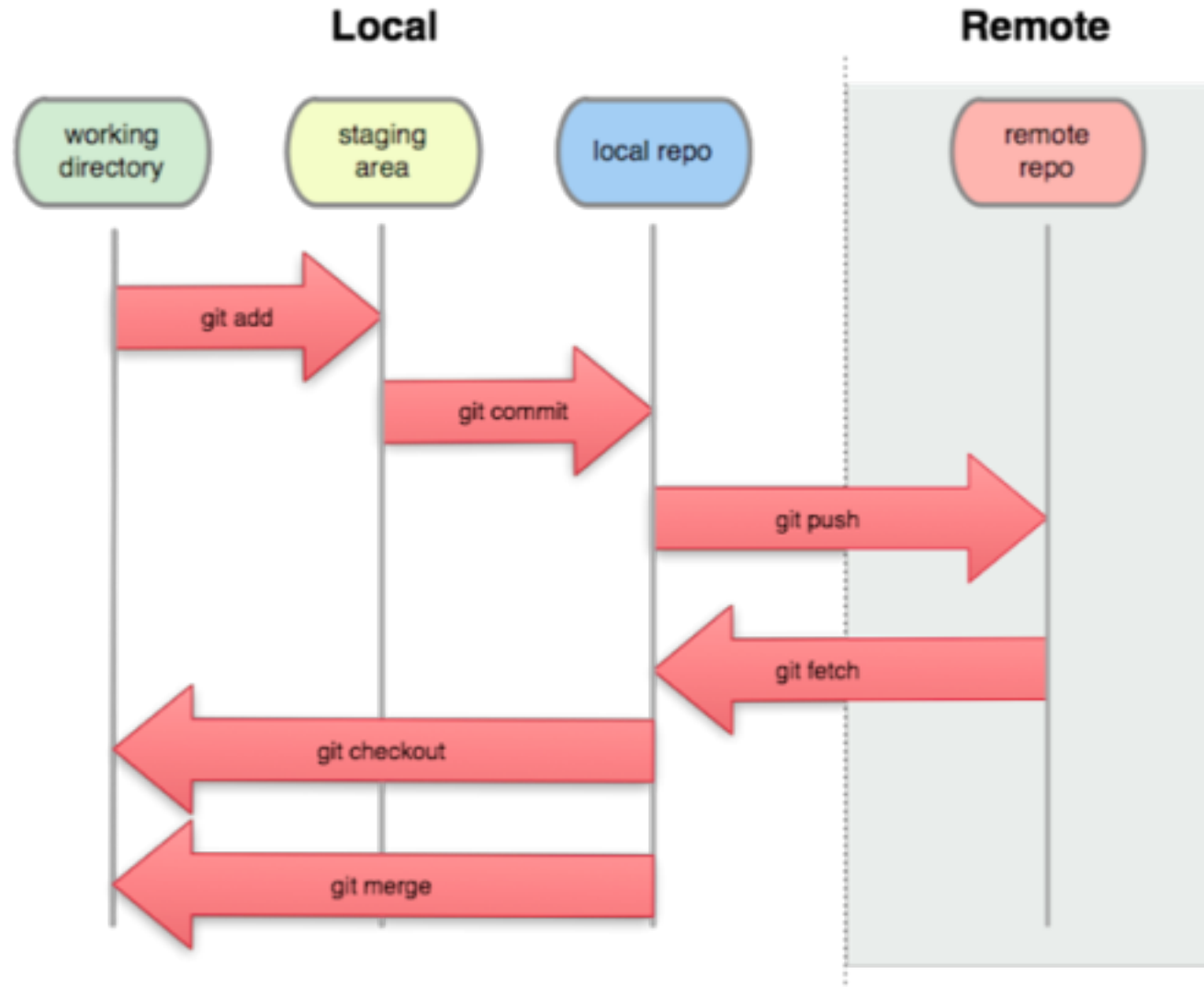
# How Git works?

## Staging directory



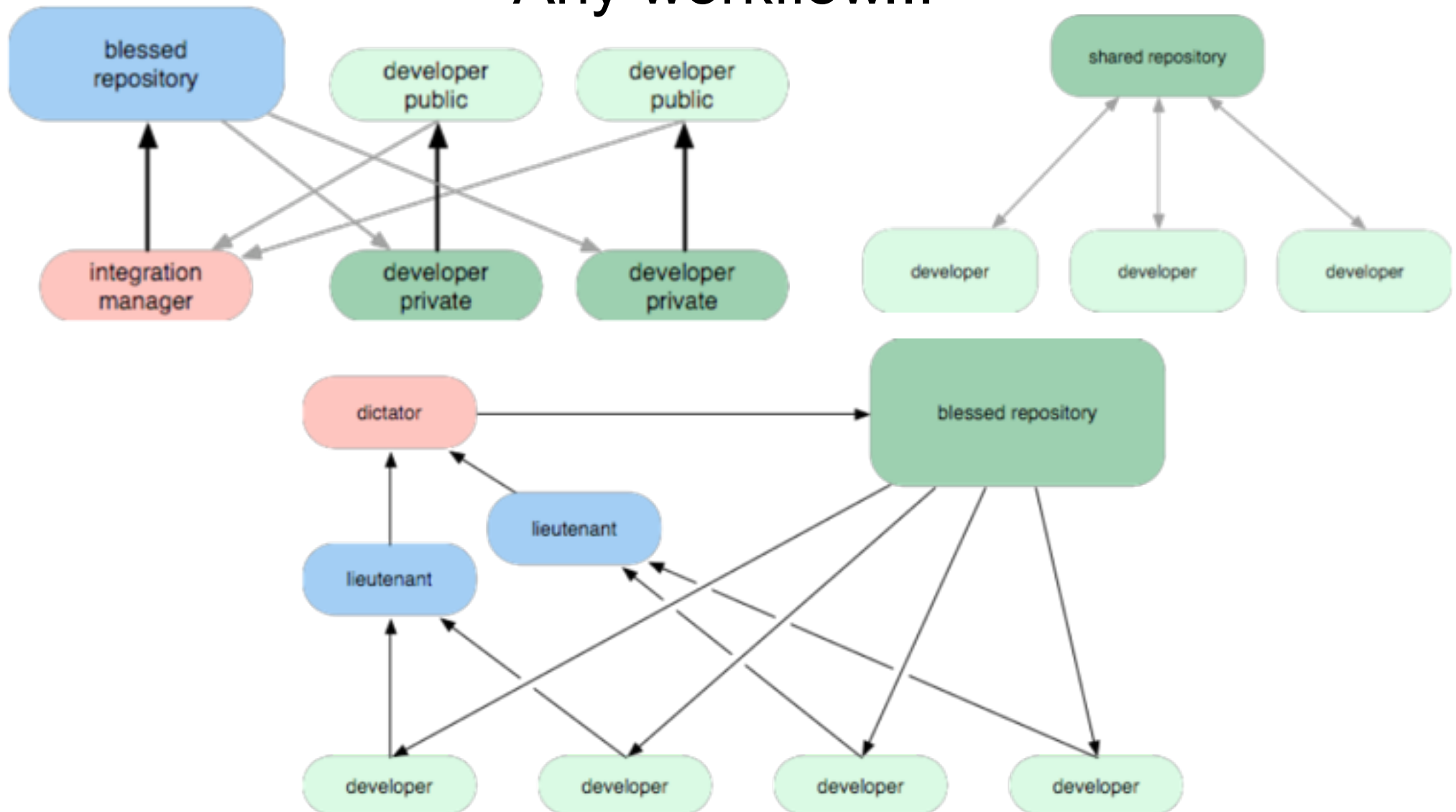
# How Git works?

## Typical setup



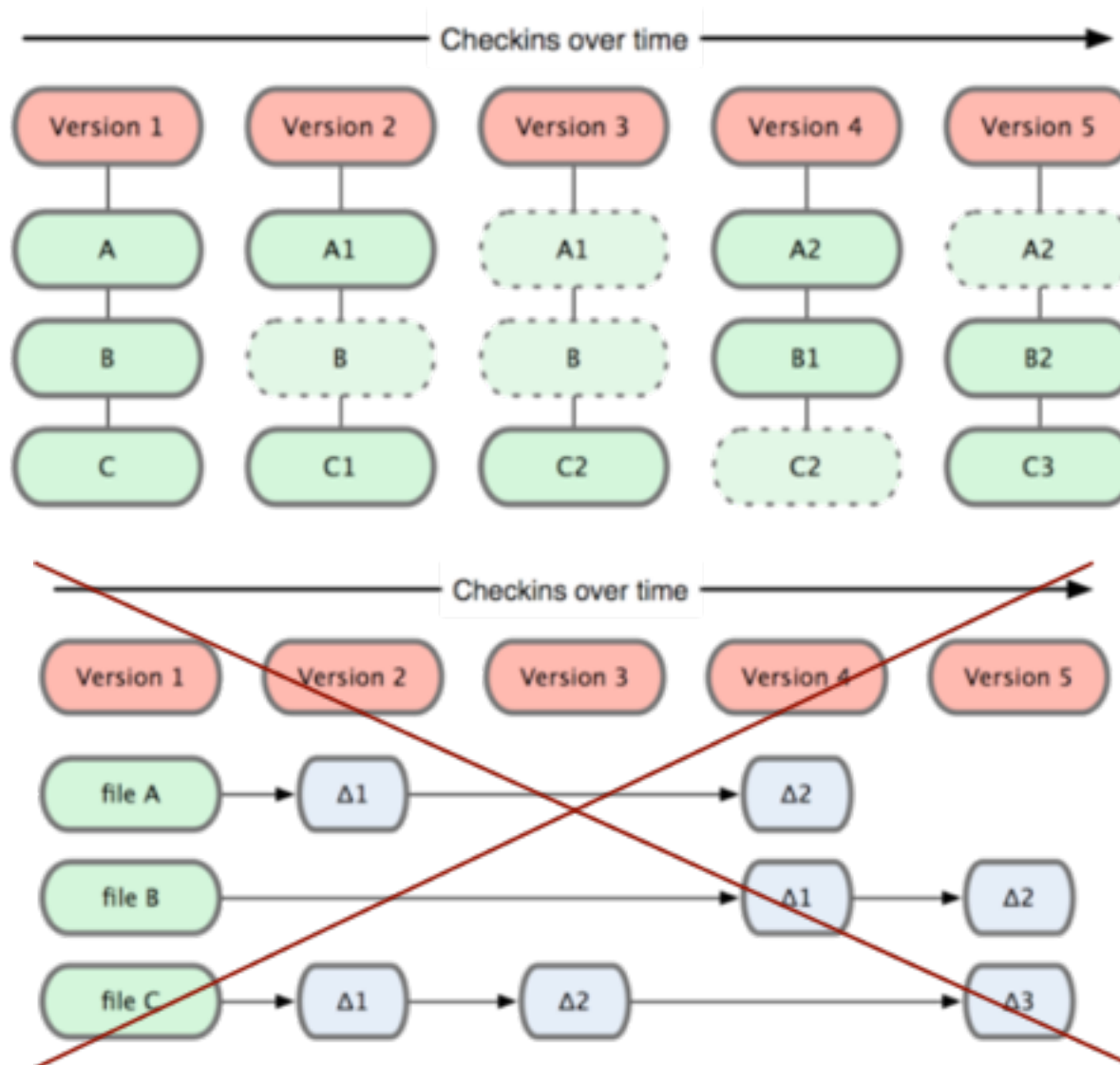
# How Git works?

Any workflow...



# How Git works?

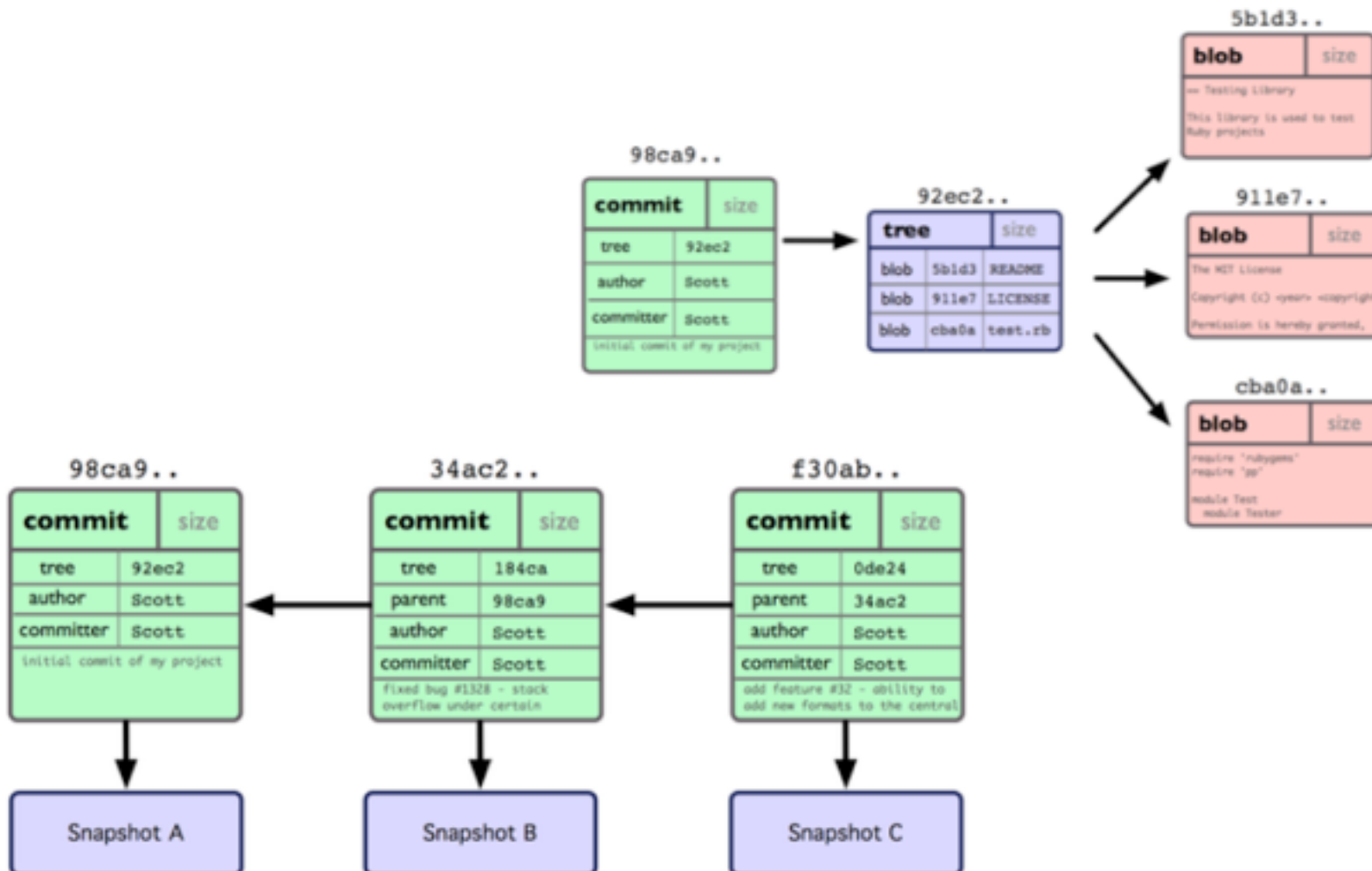
## Saves state not deltas.



Images taken from Pro Git book (<http://progit.org>)

# How Git works?

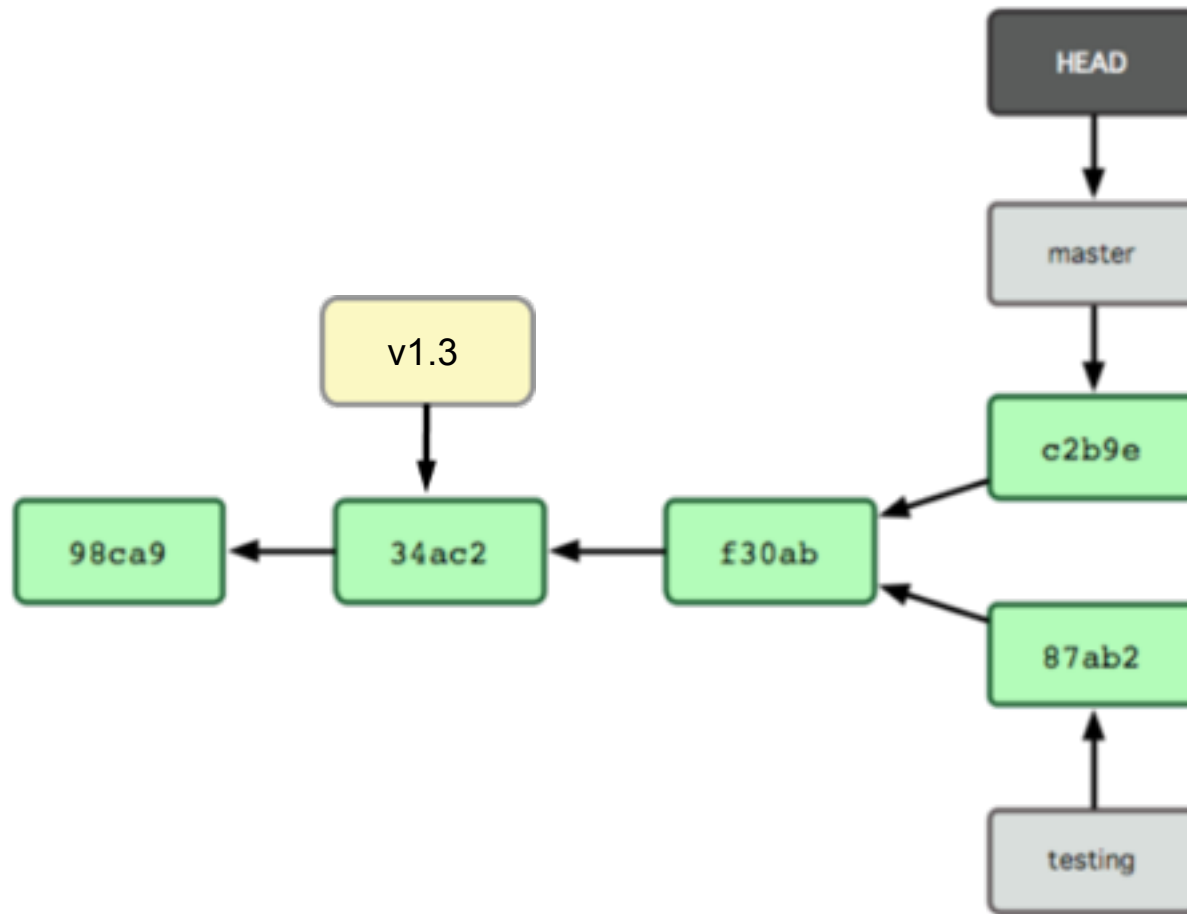
## Objects





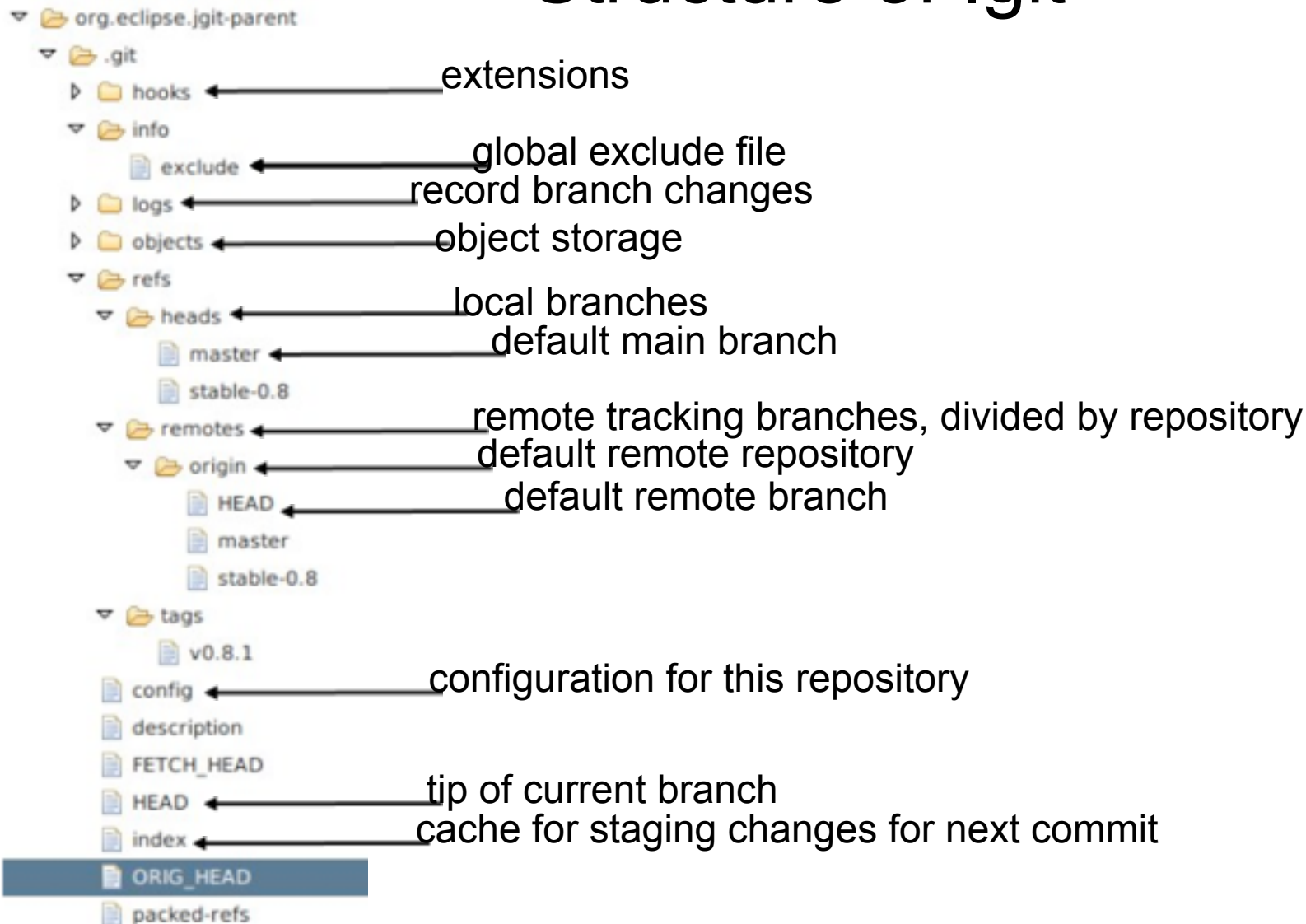
# How Git works?

## Branches and tags



# How Git works?

## Structure of .git



# Merging



Merges are just the weaving together of two (or more) local branches into one

However, unlike CVCS, you don't have to specify anything about where you're merging from and to; the trees automatically know what their split point was in the past, and can work it out from there.

Merging is much easier in a DVCS like Git

# Agenda



- Introduction
- How Git Works
- **Demos**
- Conclusion
- Q&A

# Demo

A tour of EGit  
and

Gerrit Code Review

“How the EGit and JGit teams work”

# Agenda



- Introduction
- How Git Works
- Demos
- **Conclusion**
- Q&A

# Conclusion



DVCS like Git are powerful

Git supports convenient branching and merging

Git is very fast and scales well

Gerrit enables a nice code review workflow

Git is the future SCM of Eclipse

# Resources



Ask questions on the EGit/JGit forums

<http://git-scm.com/documentation> is your friend

If you want comedy, watch Linus' talk at Google:  
<http://www.youtube.com/watch?v=4XpnKHJAok8>

Read Pro Git: <http://progit.org/book/>

<http://whygitisbetterthanx.com/>

Read the EGit User Guide

[http://wiki.eclipse.org/EGit/User\\_Guide](http://wiki.eclipse.org/EGit/User_Guide)





# Thanks!

Q & A