Git at Eclipse
http://eclipse.org/egit
Chris Aniszczyk (Red Hat)
zx@redhat.com
http://aniszczyk.org

Shawn Pearce (Google)
sop@google.com
http://spearce.org
Agenda

- Introduction
- How Git Works
- Demos
  - A tour of EGit
  - Gerrit workflow: How the EGit and JGit teams work
- 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)
- + Darius Luzska (Independent)
- + Benjamin Muskalla (EclipseSource)
- 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
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)
        Full Merge, Synchronize View, Stashing, .gitignore

12/2010 Planned 1.0 and stable API
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

- Supported
- Partially supported
- Not yet supported

- git init / git clone
- git add
- git status
- git commit
- git diff
- git fetch
- git log
- git merge*
- git rebase
- git remote

- git pull
- git push
- git stash*
- git branch
- git tag
- git checkout
- git config*
- git format-patch
- git mv* / git rm
- git reset

* planned for 0.9 in September ’10
Planned work for 0.9 (Sept 2010)

- **EGit**
  - Synchronize view
  - Changesets
  - Full merge support
  - Usability polish
- **JGit**
  - git-notes support
  - Git API improvements
No Free Lunch

The best way to learn Git is to use Git
Git vs. CVS/SVN

● Distributed (git)
  o Full local history
  o Work off-line
  o Fast
  o Rebase patches easily
  o Forks happen, deal with it
  o 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

Images taken from Pro Git book (http://progit.org)
How Git works?

Typical setup

Images taken from Pro Git book (http://progit.org)
How Git works?

Any workflow...

Images taken from Pro Git book (http://progit.org)
How Git works?
Saves state not deltas.

Images taken from Pro Git book (http://progit.org)
How Git works?

Objects

Images taken from Pro Git book (http://progit.org)
How Git works?

Branches and tags

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

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
Thanks!

Q & A