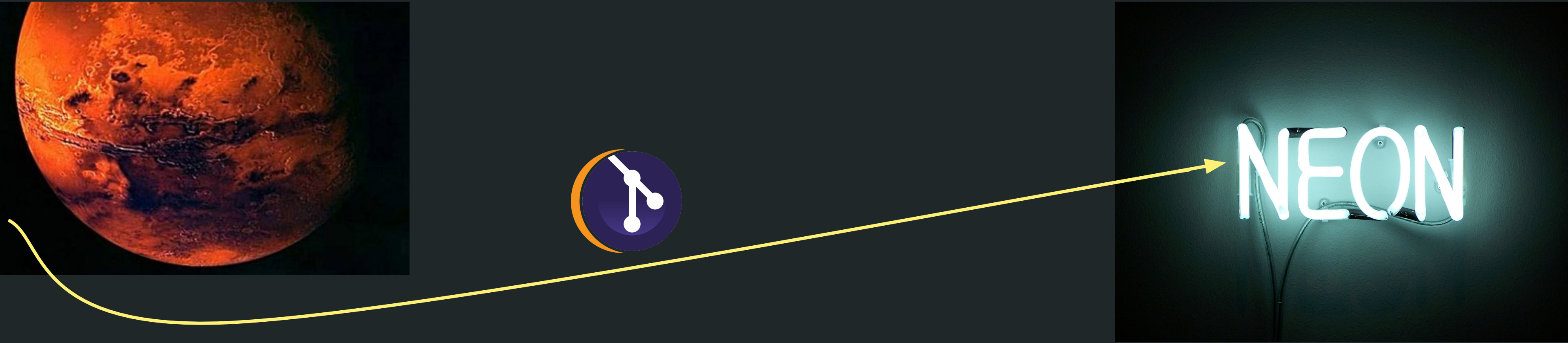


Git Journey from Mars to Neon

Matthias Sohn (SAP)

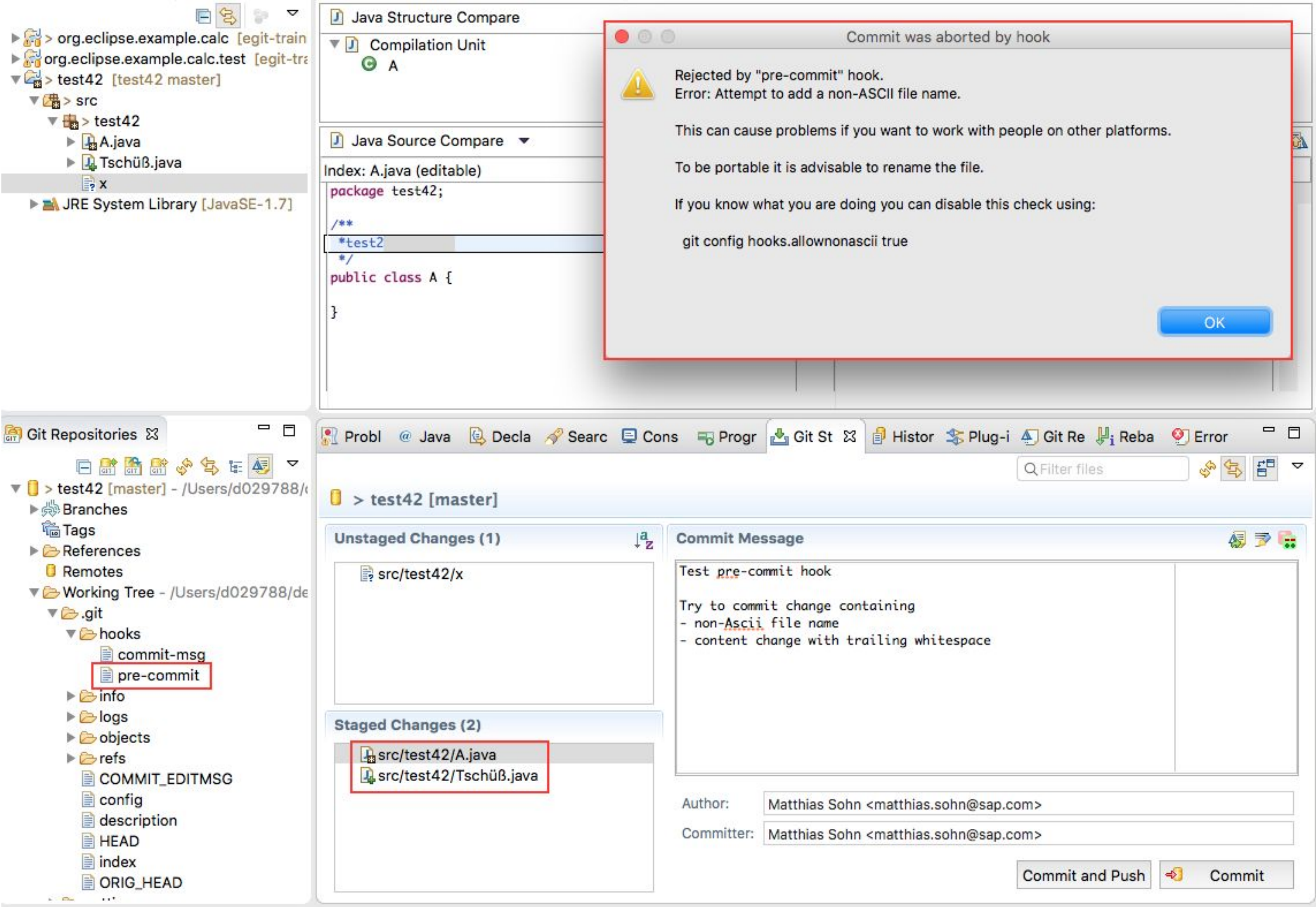
hooks support

JGit

- pre-commit (4.0)
- commit-msg (4.0)
- pre-push (4.2)

Hooks

- pre-commit
- commit-msg
- pre-push



```
package test42;

/**
 *test2
 */
public class A {

}
```

Commit was aborted by hook

Rejected by "pre-commit" hook.
Error: Attempt to add a non-ASCII file name.

This can cause problems if you want to work with people on other platforms.

To be portable it is advisable to rename the file.

If you know what you are doing you can disable this check using:

```
git config hooks.allownonascii true
```

OK

```
Test pre-commit hook

Try to commit change containing
- non-Ascii file name
- content change with trailing whitespace
```

Author: Matthias Sohn <matthias.sohn@sap.com>

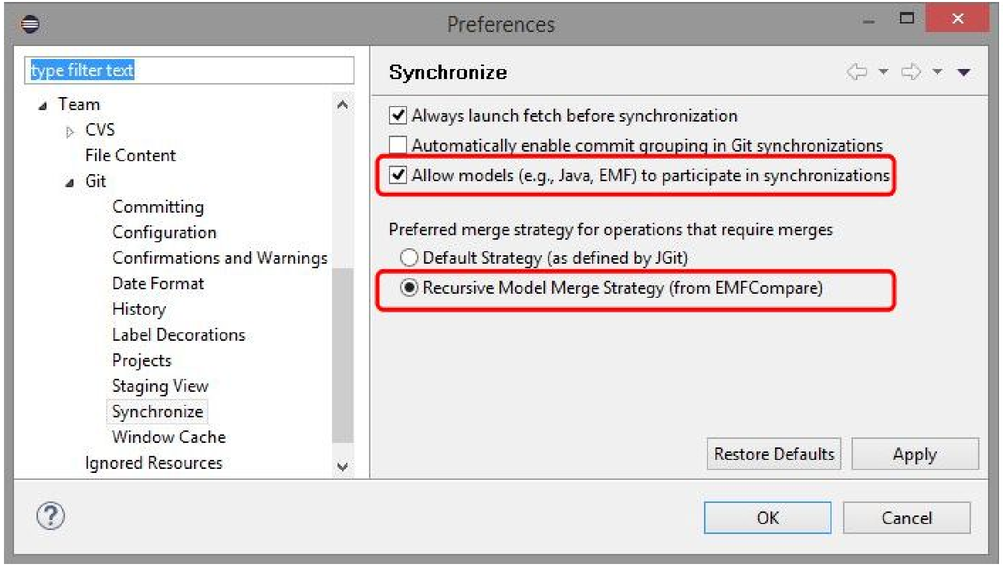
Committer: Matthias Sohn <matthias.sohn@sap.com>

merge strategy extension point

JGit 4.0

- allows to plugin external merge strategy
- implemented by EMF Compare to provide model merge strategy

EMF Compare provides model merge strategy



Preferences

type filter text

- Team
  - CVS
  - File Content
  - Git
    - Committing
    - Configuration
    - Confirmations and Warnings
    - Date Format
    - History
    - Label Decorations
    - Projects
    - Staging View
    - Synchronize
    - Window Cache
  - Ignored Resources

Synchronize

☑ Always launch fetch before synchronization

□ Automatically enable commit grouping in Git synchronizations

☑ Allow models (e.g., Java, EMF) to participate in synchronizations

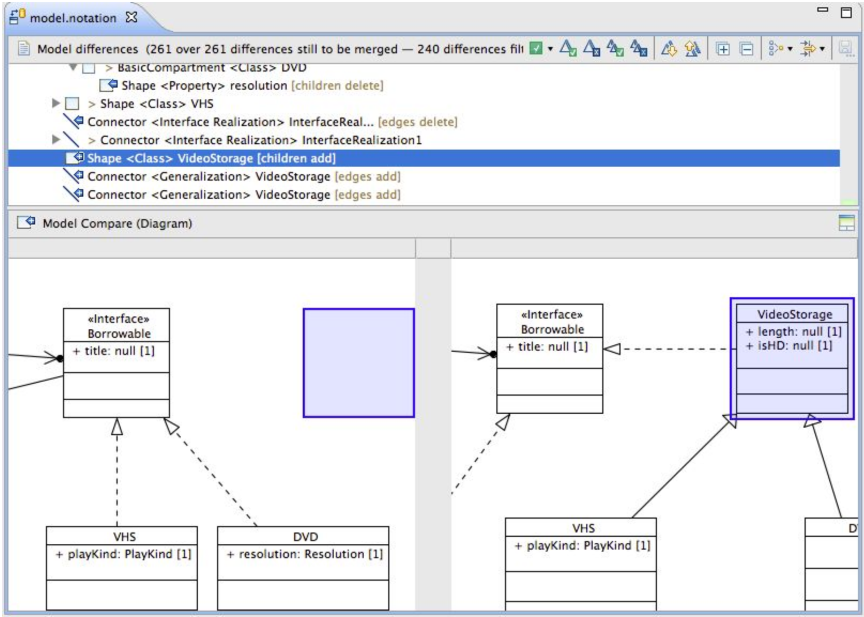
Preferred merge strategy for operations that require merges

○ Default Strategy (as defined by JGit)

● Recursive Model Merge Strategy (from EMFCompare)

Computes the logical model for each version

Compares / merges complete logical model



model.notation

Model differences (261 over 261 differences still to be merged — 240 differences filt

```
    > BasicCompartment <Class> DVD
      Shape <Property> resolution [children delete]
  > Shape <Class> VHS
  Connector <Interface Realization> InterfaceReal... [edges delete]
  > Connector <Interface Realization> InterfaceRealization1
  Shape <Class> VideoStorage [children add]
  Connector <Generalization> VideoStorage [edges add]
  Connector <Generalization> VideoStorage [edges add]
```

Model Compare (Diagram)

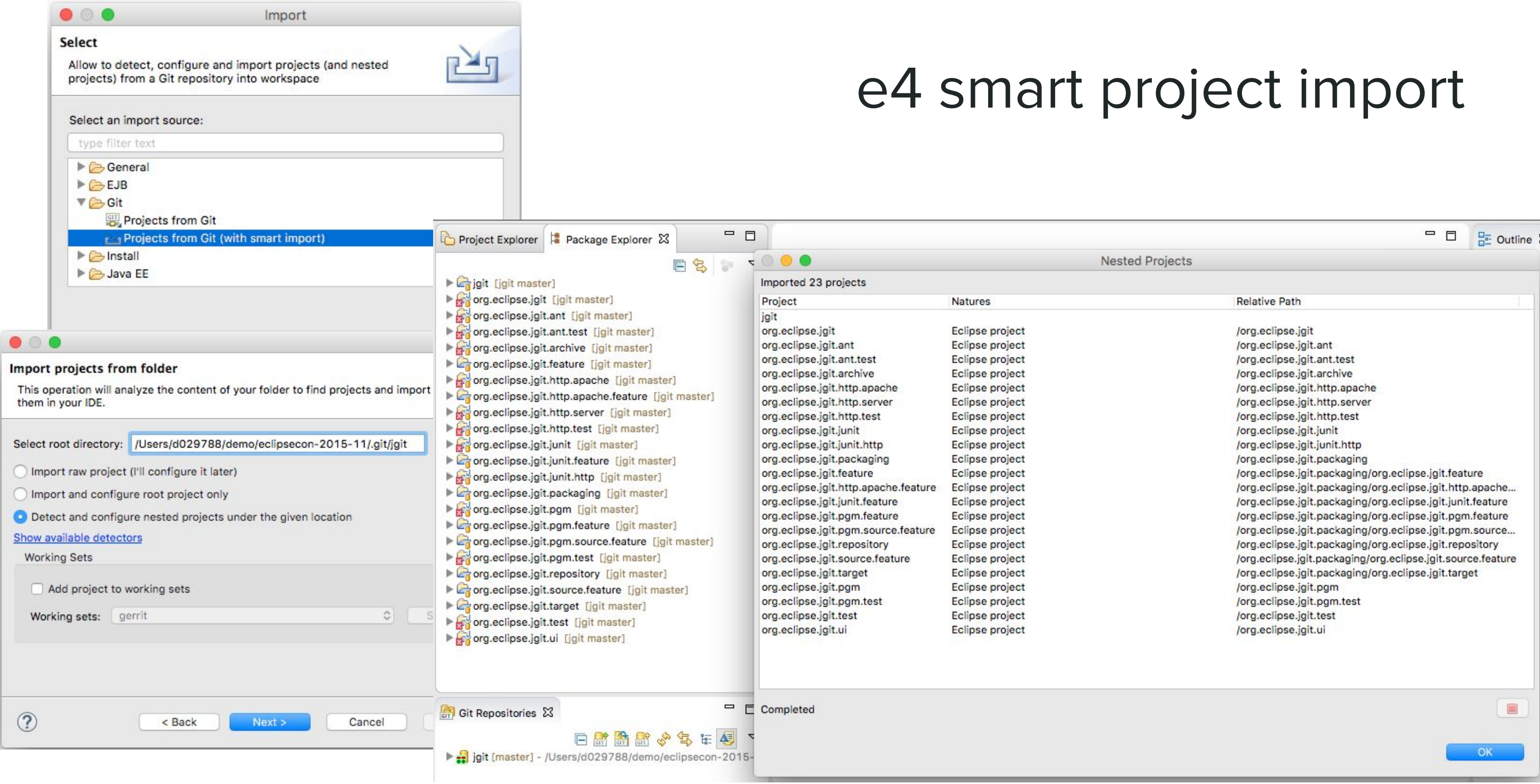
e4 smart project import wizard

EGit 4.0

- clone repository
- invoke e4 smart project import
- auto-detects projects and their project types

wiki e4 smart import

e4 smart project import



Select

Allow to detect, configure and import projects (and nested projects) from a Git repository into workspace

Select an import source:

type filter text

- ▶ General
- ▶ EJB
- ▼ Git
  - Projects from Git
  - Projects from Git (with smart import)
- ▶ Install
- ▶ Java EE

Import projects from folder

This operation will analyze the content of your folder to find projects and import them in your IDE.

Select root directory: /Users/d029788/demo/eclipsecon-2015-11/.git/jgit

○ Import raw project (I'll configure it later)

○ Import and configure root project only

● Detect and configure nested projects under the given location

Show available detectors

Working Sets

□ Add project to working sets

Working sets: gerrit

- ▶ jgit [jgit master]
- ▶ org.eclipse.jgit [jgit master]
- ▶ org.eclipse.jgit.ant [jgit master]
- ▶ org.eclipse.jgit.ant.test [jgit master]
- ▶ org.eclipse.jgit.archive [jgit master]
- ▶ org.eclipse.jgit.feature [jgit master]
- ▶ org.eclipse.jgit.http.apache [jgit master]
- ▶ org.eclipse.jgit.http.apache.feature [jgit master]
- ▶ org.eclipse.jgit.http.server [jgit master]
- ▶ org.eclipse.jgit.http.test [jgit master]
- ▶ org.eclipse.jgit.junit [jgit master]
- ▶ org.eclipse.jgit.junit.feature [jgit master]
- ▶ org.eclipse.jgit.junit.http [jgit master]
- ▶ org.eclipse.jgit.packaging [jgit master]
- ▶ org.eclipse.jgit.pgm [jgit master]
- ▶ org.eclipse.jgit.pgm.feature [jgit master]
- ▶ org.eclipse.jgit.pgm.source.feature [jgit master]
- ▶ org.eclipse.jgit.pgm.test [jgit master]
- ▶ org.eclipse.jgit.repository [jgit master]
- ▶ org.eclipse.jgit.source.feature [jgit master]
- ▶ org.eclipse.jgit.target [jgit master]
- ▶ org.eclipse.jgit.test [jgit master]
- ▶ org.eclipse.jgit.ui [jgit master]

- ▶ jgit [master] - /Users/d029788/demo/eclipsecon-2015-

Imported 23 projects

Project	Natures	Relative Path
jgit		
org.eclipse.jgit	Eclipse project	/org.eclipse.jgit
org.eclipse.jgit.ant	Eclipse project	/org.eclipse.jgit.ant
org.eclipse.jgit.ant.test	Eclipse project	/org.eclipse.jgit.ant.test
org.eclipse.jgit.archive	Eclipse project	/org.eclipse.jgit.archive
org.eclipse.jgit.http.apache	Eclipse project	/org.eclipse.jgit.http.apache
org.eclipse.jgit.http.server	Eclipse project	/org.eclipse.jgit.http.server
org.eclipse.jgit.http.test	Eclipse project	/org.eclipse.jgit.http.test
org.eclipse.jgit.junit	Eclipse project	/org.eclipse.jgit.junit
org.eclipse.jgit.junit.http	Eclipse project	/org.eclipse.jgit.junit.http
org.eclipse.jgit.packaging	Eclipse project	/org.eclipse.jgit.packaging
org.eclipse.jgit.feature	Eclipse project	/org.eclipse.jgit.packaging/org.eclipse.jgit.feature
org.eclipse.jgit.http.apache.feature	Eclipse project	/org.eclipse.jgit.packaging/org.eclipse.jgit.http.apache...
org.eclipse.jgit.junit.feature	Eclipse project	/org.eclipse.jgit.packaging/org.eclipse.jgit.junit.feature
org.eclipse.jgit.pgm.feature	Eclipse project	/org.eclipse.jgit.packaging/org.eclipse.jgit.pgm.feature
org.eclipse.jgit.pgm.source.feature	Eclipse project	/org.eclipse.jgit.packaging/org.eclipse.jgit.pgm.source...
org.eclipse.jgit.repository	Eclipse project	/org.eclipse.jgit.packaging/org.eclipse.jgit.repository
org.eclipse.jgit.source.feature	Eclipse project	/org.eclipse.jgit.packaging/org.eclipse.jgit.source.feature
org.eclipse.jgit.target	Eclipse project	/org.eclipse.jgit.packaging/org.eclipse.jgit.target
org.eclipse.jgit.pgm	Eclipse project	/org.eclipse.jgit.pgm
org.eclipse.jgit.pgm.test	Eclipse project	/org.eclipse.jgit.pgm.test
org.eclipse.jgit.test	Eclipse project	/org.eclipse.jgit.test
org.eclipse.jgit.ui	Eclipse project	/org.eclipse.jgit.ui

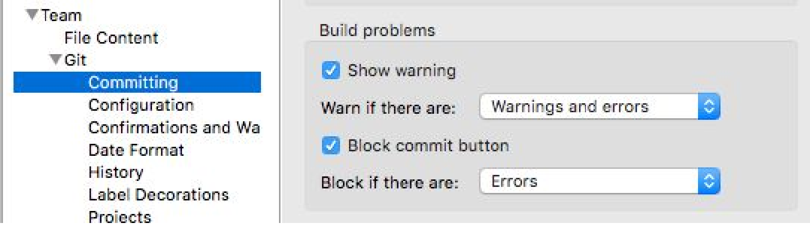
Completed

Staging View

EGit

- block commit on errors (4.1)
- compare mode (4.1)
- better support non-workspace files (4.2)
- enable sorting files by status (4.3)

Warn/block commit on errors



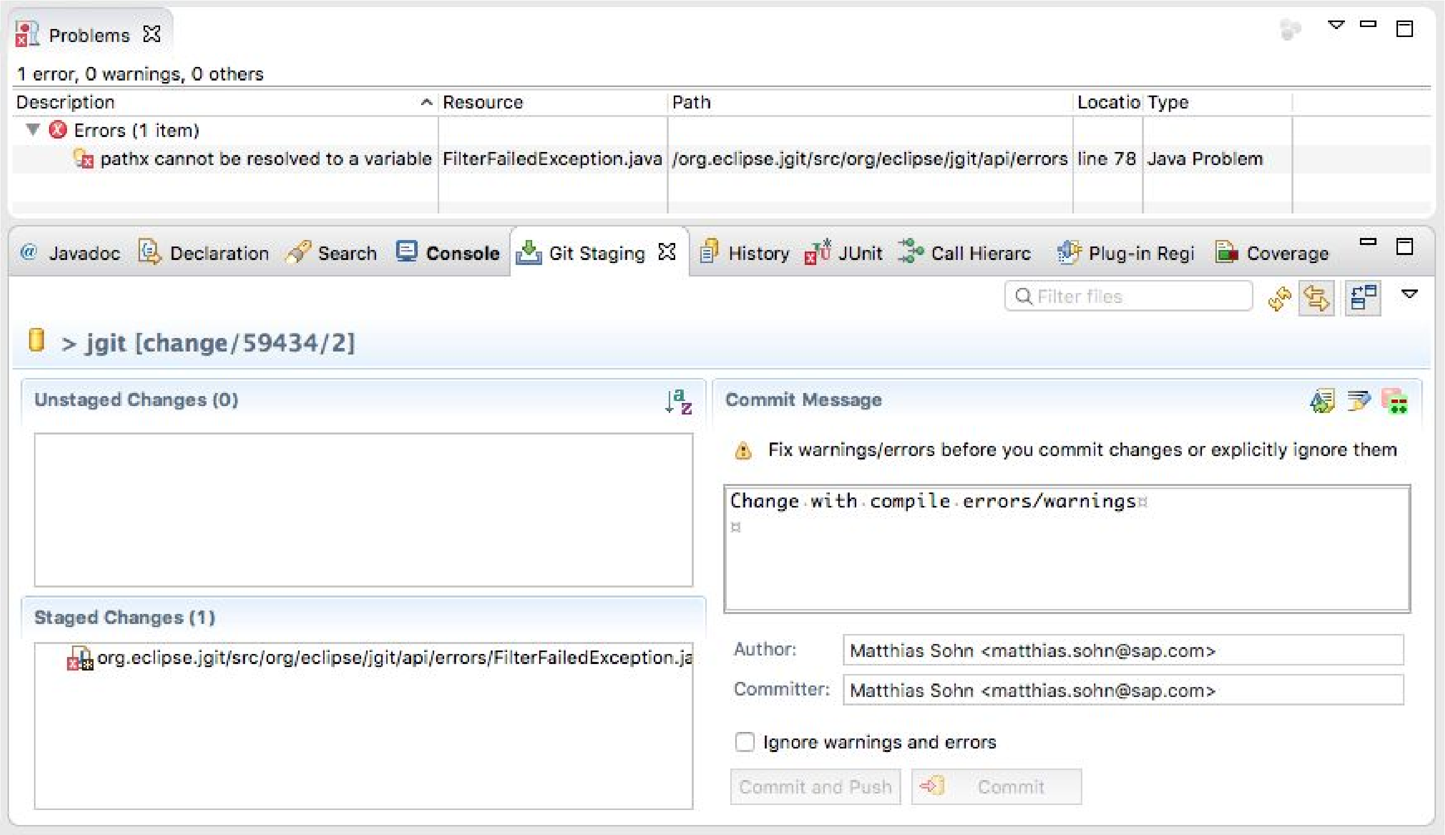
Build problems

☑ Show warning

Warn if there are: Warnings and errors

☑ Block commit button

Block if there are: Errors



1 error, 0 warnings, 0 others

Description	Resource	Path	Locatio	Type
Errors (1 item)				
pathx cannot be resolved to a variable	FilterFailedException.java	/org.eclipse.jgit/src/org/eclipse/jgit/api/errors	line 78	Java Problem

> jgit [change/59434/2]

Unstaged Changes (0)

Staged Changes (1)

org.eclipse.jgit/src/org/eclipse/jgit/api/errors/FilterFailedException.ja

Fix warnings/errors before you commit changes or explicitly ignore them

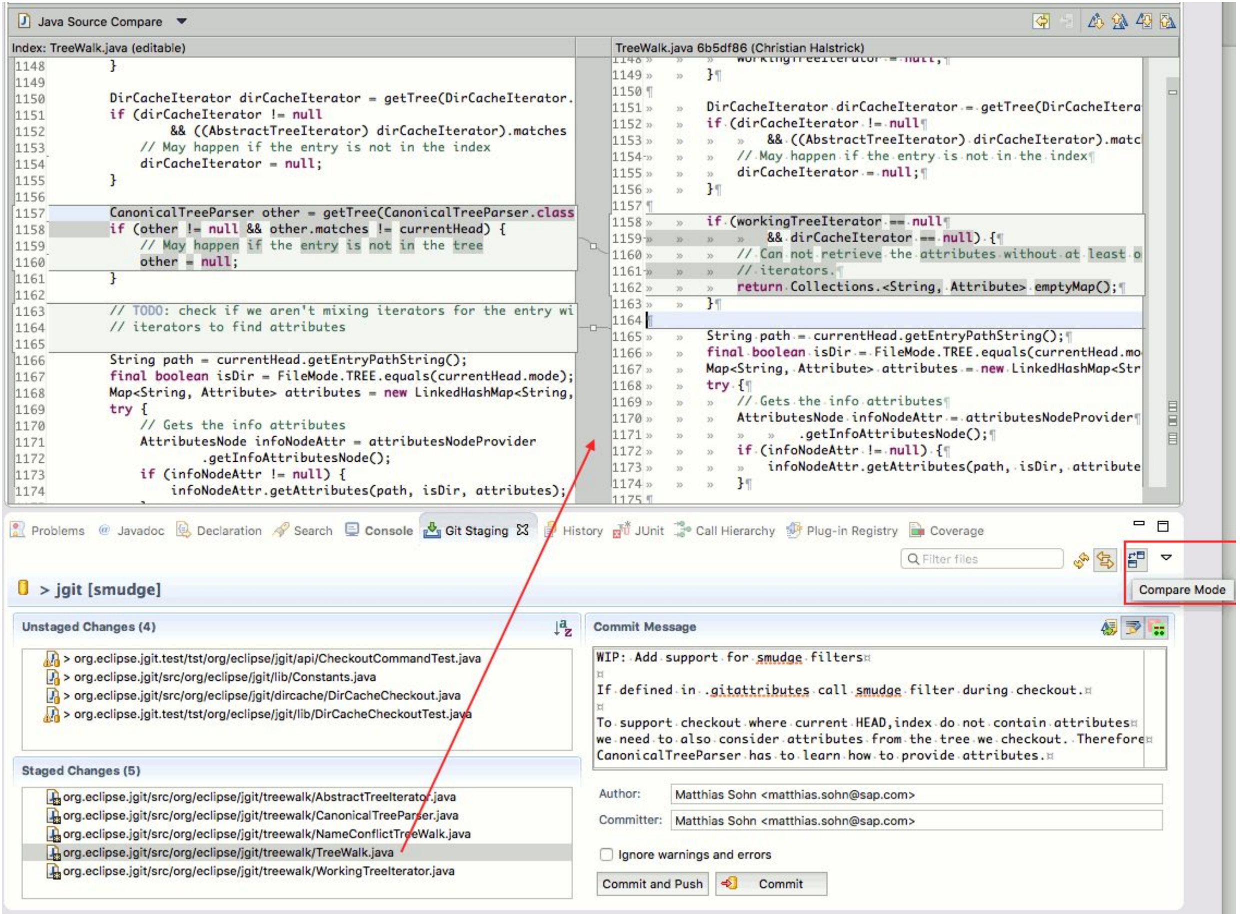
```
Change with compile errors/warnings
```

Author: Matthias Sohn <matthias.sohn@sap.com>

Committer: Matthias Sohn <matthias.sohn@sap.com>

□ Ignore warnings and errors

Compare Mode in Staging View



Index: TreeWalk.java (editable)

```java
		}

		DirCacheIterator dirCacheIterator = getTree(DirCacheIterator.
		if (dirCacheIterator != null
				&& ((AbstractTreeIterator) dirCacheIterator).matches
			// May happen if the entry is not in the index
			dirCacheIterator = null;
		}

		CanonicalTreeParser other = getTree(CanonicalTreeParser.class
		if (other != null && other.matches != currentHead) {
			// May happen if the entry is not in the tree
			other = null;
		}

		// TODO: check if we aren't mixing iterators for the entry wi
		// iterators to find attributes

		String path = currentHead.getEntryPathString();
		final boolean isDir = FileMode.TREE.equals(currentHead.mode);
		Map<String, Attribute> attributes = new LinkedHashMap<String,
		try {
			// Gets the info attributes
			AttributesNode infoNodeAttr = attributesNodeProvider
					.getInfoAttributesNode();
			if (infoNodeAttr != null) {
				infoNodeAttr.getAttributes(path, isDir, attributes);
```

TreeWalk.java 6b5df86 (Christian Halstrick)

```java
			workingTreeIterator = null;
		}

		DirCacheIterator dirCacheIterator = getTree(DirCacheItera
		if (dirCacheIterator != null
				&& ((AbstractTreeIterator) dirCacheIterator).matc
			// May happen if the entry is not in the index
			dirCacheIterator = null;
		}

		if (workingTreeIterator == null
				&& dirCacheIterator == null) {
			// Can not retrieve the attributes without at least o
			// iterators.
			return Collections.<String, Attribute> emptyMap();
		}

		String path = currentHead.getEntryPathString();
		final boolean isDir = FileMode.TREE.equals(currentHead.mo
		Map<String, Attribute> attributes = new LinkedHashMap<Str
		try {
			// Gets the info attributes
			AttributesNode infoNodeAttr = attributesNodeProvider
					.getInfoAttributesNode();
			if (infoNodeAttr != null) {
				infoNodeAttr.getAttributes(path, isDir, attribute
			}

```

> jgit [smudge]

Unstaged Changes (4)

- > org.eclipse.jgit.test/tst/org/eclipse/jgit/api/CheckoutCommandTest.java
- > org.eclipse.jgit/src/org/eclipse/jgit/lib/Constants.java
- > org.eclipse.jgit/src/org/eclipse/jgit/dircache/DirCacheCheckout.java
- > org.eclipse.jgit.test/tst/org/eclipse/jgit/lib/DirCacheCheckoutTest.java

Staged Changes (5)

- org.eclipse.jgit/src/org/eclipse/jgit/treewalk/AbstractTreeIterator.java
- org.eclipse.jgit/src/org/eclipse/jgit/treewalk/CanonicalTreeParser.java
- org.eclipse.jgit/src/org/eclipse/jgit/treewalk/NameConflictTreeWalk.java
- org.eclipse.jgit/src/org/eclipse/jgit/treewalk/TreeWalk.java
- org.eclipse.jgit/src/org/eclipse/jgit/treewalk/WorkingTreeIterator.java

Commit Message

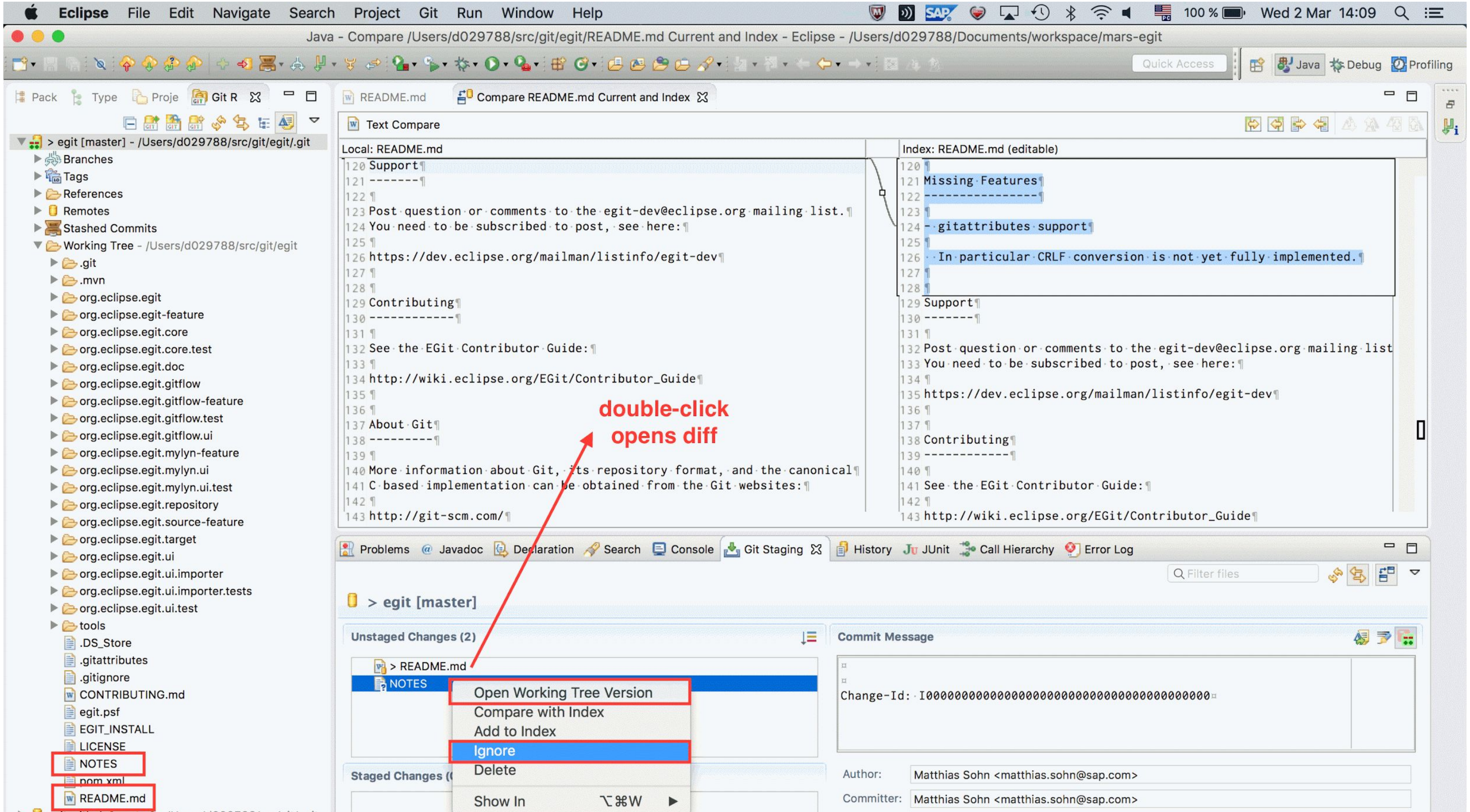
```
WIP: Add support for smudge filters

If defined in .gitattributes call smudge filter during checkout.

To support checkout where current HEAD,index do not contain attributes
we need to also consider attributes from the tree we checkout. Therefore
CanonicalTreeParser has to learn how to provide attributes.
```

Author: Matthias Sohn <matthias.sohn@sap.com>

Committer: Matthias Sohn <matthias.sohn@sap.com>

□ Ignore warnings and errors

Better support non-workspace files



Java - Compare /Users/d029788/src/git/egit/README.md Current and Index - Eclipse - /Users/d029788/Documents/workspace/mars-egit

- > egit [master] - /Users/d029788/src/git/egit/.git
  - Branches
  - Tags
  - References
  - Remotes
  - Stashed Commits
  - Working Tree - /Users/d029788/src/git/egit
    - .git
    - .mvn
    - org.eclipse.egit
    - org.eclipse.egit-feature
    - org.eclipse.egit.core
    - org.eclipse.egit.core.test
    - org.eclipse.egit.doc
    - org.eclipse.egit.gitflow
    - org.eclipse.egit.gitflow-feature
    - org.eclipse.egit.gitflow.test
    - org.eclipse.egit.gitflow.ui
    - org.eclipse.egit.mylyn-feature
    - org.eclipse.egit.mylyn.ui
    - org.eclipse.egit.mylyn.ui.test
    - org.eclipse.egit.repository
    - org.eclipse.egit.source-feature
    - org.eclipse.egit.target
    - org.eclipse.egit.ui
    - org.eclipse.egit.ui.importer
    - org.eclipse.egit.ui.importer.tests
    - org.eclipse.egit.ui.test
    - tools
    - .DS_Store
    - .gitattributes
    - .gitignore
    - CONTRIBUTING.md
    - egit.psf
    - EGIT_INSTALL
    - LICENSE
    - NOTES
    - pom.xml
    - README.md

Local: README.md

```
Support
-------

Post question or comments to the egit-dev@eclipse.org mailing list.
You need to be subscribed to post, see here:

https://dev.eclipse.org/mailman/listinfo/egit-dev


Contributing
------------

See the EGit Contributor Guide:

http://wiki.eclipse.org/EGit/Contributor_Guide


About Git
---------

More information about Git, its repository format, and the canonical
C based implementation can be obtained from the Git websites:

http://git-scm.com/
```

Index: README.md (editable)

```

Missing Features
----------------

- gitattributes support
  In particular CRLF conversion is not yet fully implemented.


Support
-------

Post question or comments to the egit-dev@eclipse.org mailing list
You need to be subscribed to post, see here:

https://dev.eclipse.org/mailman/listinfo/egit-dev


Contributing
------------

See the EGit Contributor Guide:

http://wiki.eclipse.org/EGit/Contributor_Guide
```

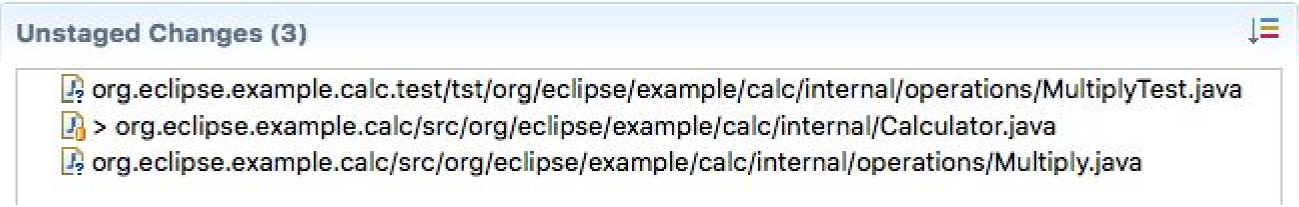
Change-Id: I0000000000000000000000000000000000000000

Author: Matthias Sohn <matthias.sohn@sap.com>

Committer: Matthias Sohn <matthias.sohn@sap.com>

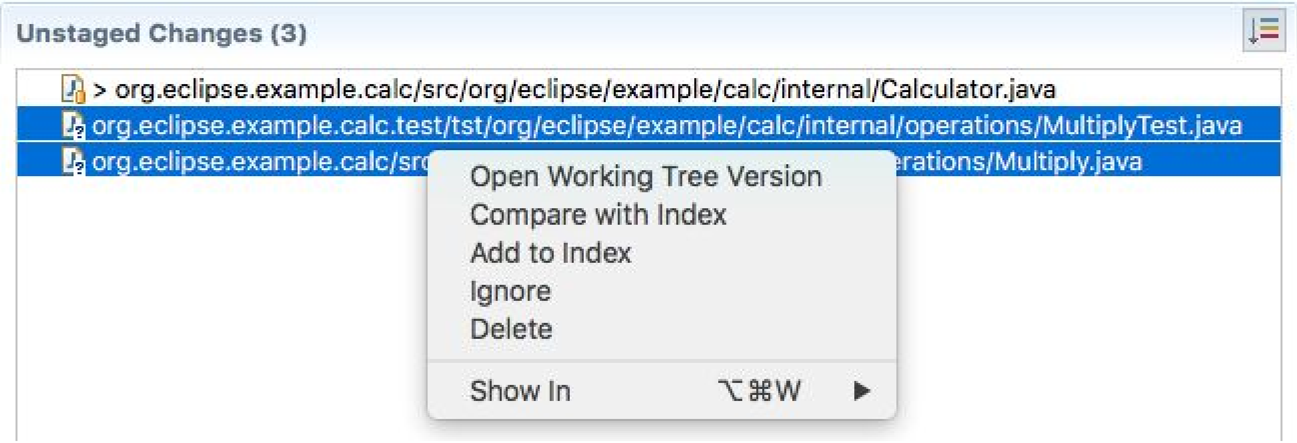
double-click opens diff

Sorting files by status



Unstaged Changes (3)

org.eclipse.example.calc.test/tst/org/eclipse/example/calc/internal/operations/MultiplyTest.java
> org.eclipse.example.calc/src/org/eclipse/example/calc/internal/Calculator.java
org.eclipse.example.calc/src/org/eclipse/example/calc/internal/operations/Multiply.java



Unstaged Changes (3)

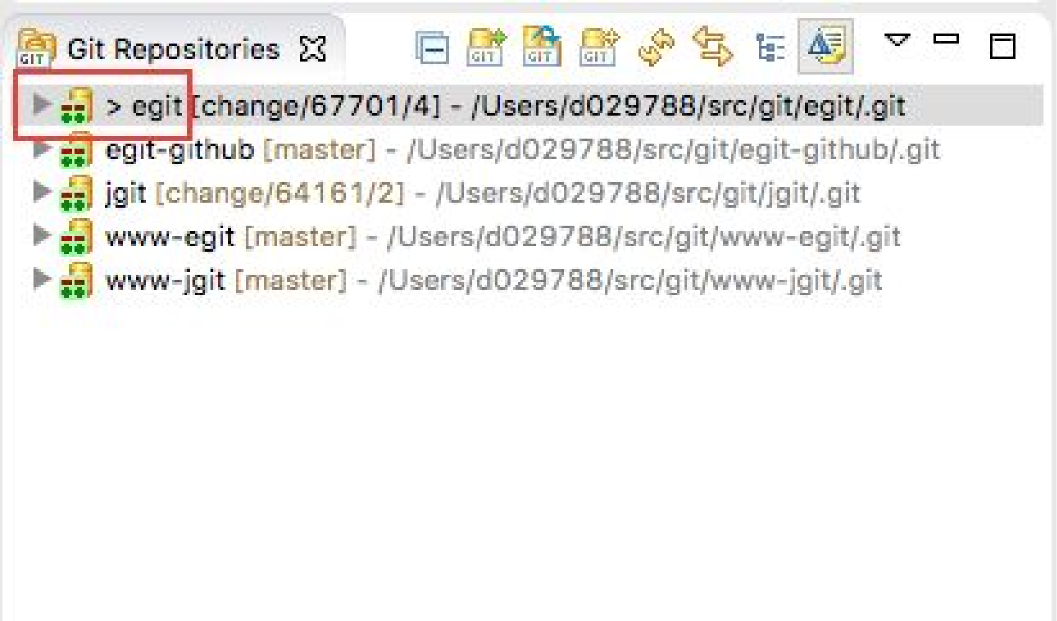
> org.eclipse.example.calc/src/org/eclipse/example/calc/internal/Calculator.java
org.eclipse.example.calc.test/tst/org/eclipse/example/calc/internal/operations/MultiplyTest.java
org.eclipse.example.calc/src erations/Multiply.java

Misc Improvements

EGit

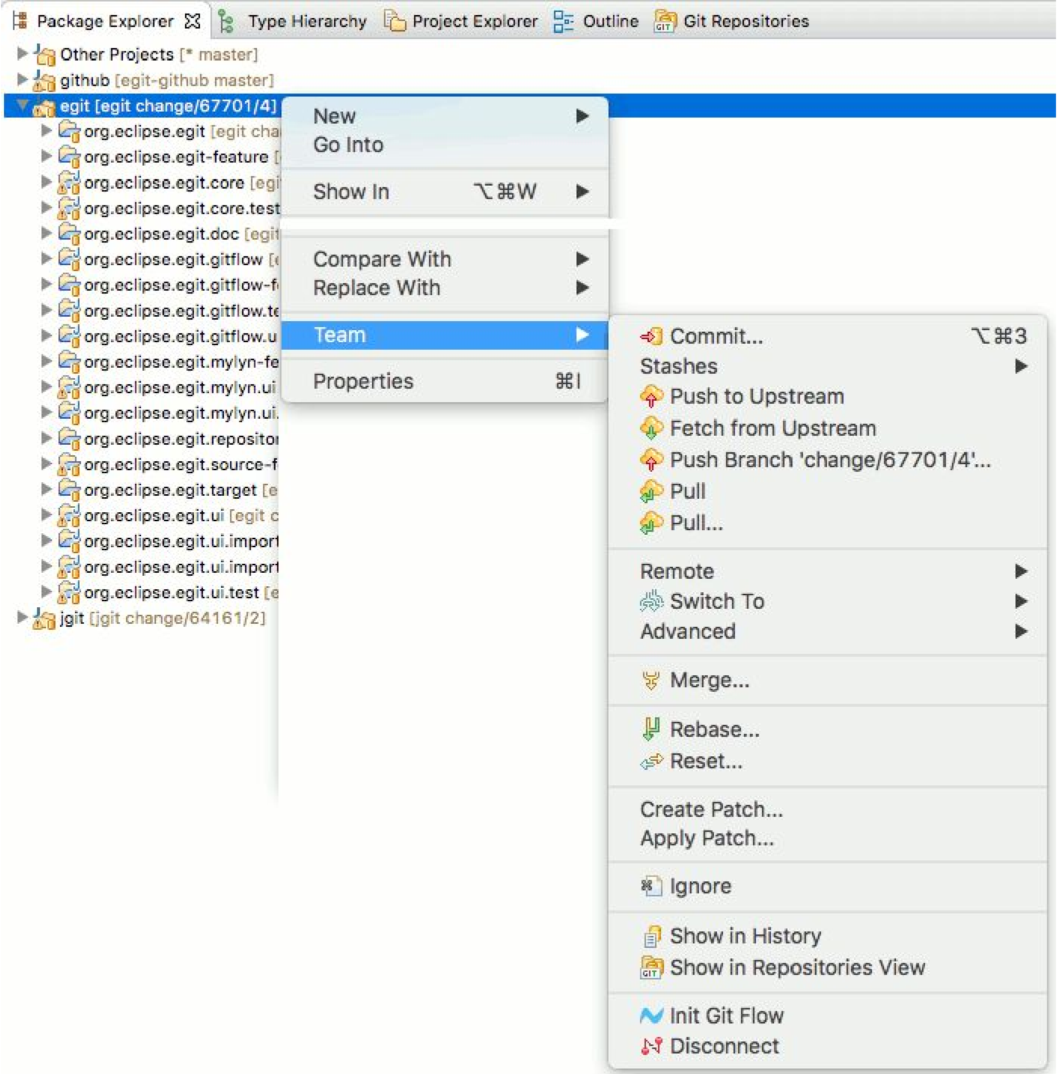
- dirty decorator in repositories view
- most git actions enabled on working sets
- configure date format
- branch auto-selection in history view
- enable symlinked .git
- support symlinks in compare editor

dirty decorator in repositories view

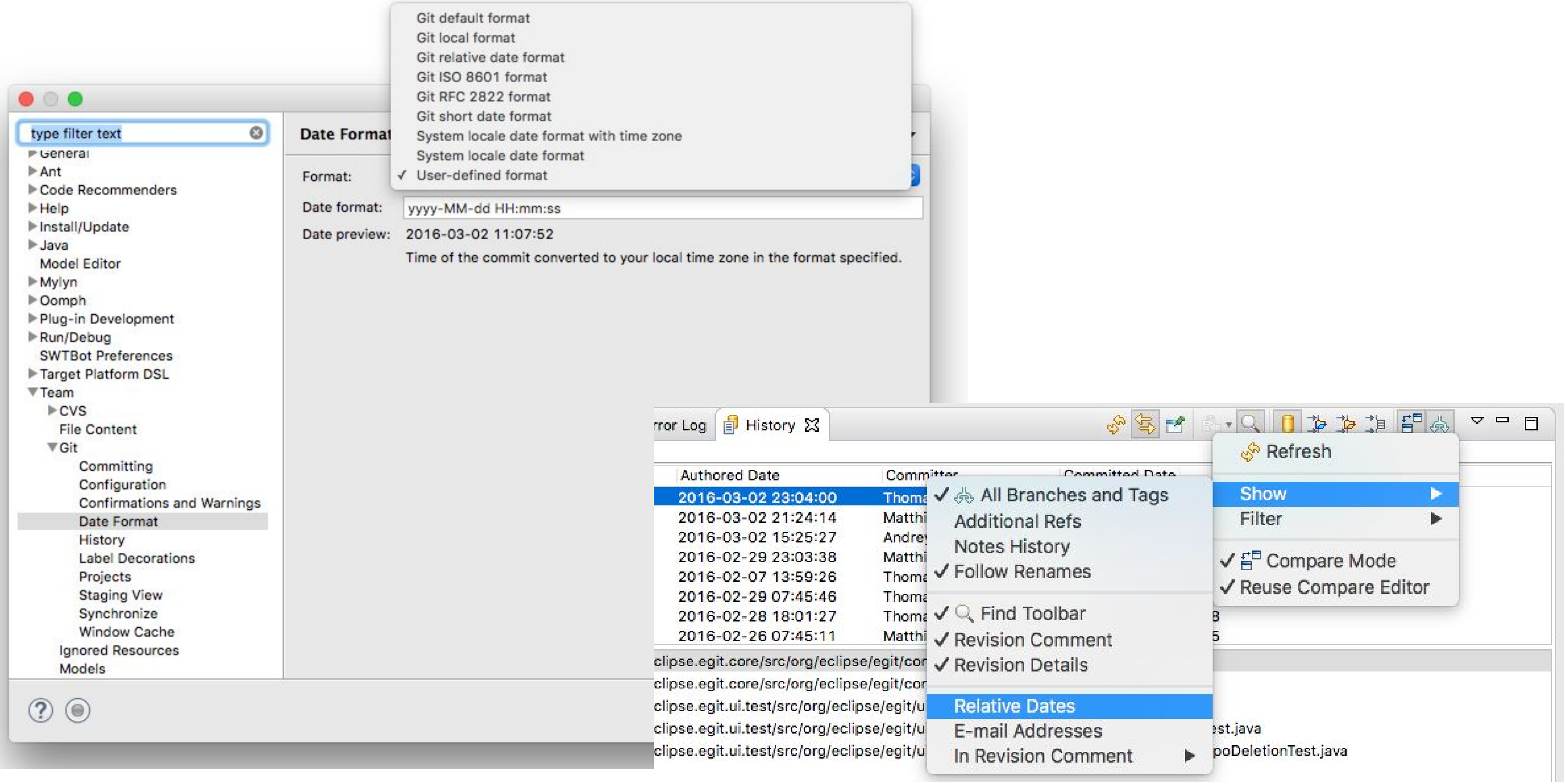


```
> egit [change/67701/4] - /Users/d029788/src/git/egit/.git
egit-github [master] - /Users/d029788/src/git/egit-github/.git
jgit [change/64161/2] - /Users/d029788/src/git/jgit/.git
www-egit [master] - /Users/d029788/src/git/www-egit/.git
www-jgit [master] - /Users/d029788/src/git/www-jgit/.git
```

enable most git actions on working sets


Package Explorer
Type Hierarchy
Project Explorer
Outline
Git Repositories
Other Projects [* master]
github [egit-github master]
egit [egit change/67701/4]
org.eclipse.egit [egit cha
org.eclipse.egit-feature [
org.eclipse.egit.core [egi
org.eclipse.egit.core.test
org.eclipse.egit.doc [egit
org.eclipse.egit.gitflow [
org.eclipse.egit.gitflow-f
org.eclipse.egit.gitflow.te
org.eclipse.egit.gitflow.u
org.eclipse.egit.mylyn-fe
org.eclipse.egit.mylyn.ui
org.eclipse.egit.mylyn.ui.
org.eclipse.egit.repositor
org.eclipse.egit.source-f
org.eclipse.egit.target [e
org.eclipse.egit.ui [egit c
org.eclipse.egit.ui.import
org.eclipse.egit.ui.import
org.eclipse.egit.ui.test [e
jgit [jgit change/64161/2]
New
Go Into
Show In ⌥⌘W
Compare With
Replace With
Team
Properties ⌘I
Commit... ⌥⌘3
Stashes
Push to Upstream
Fetch from Upstream
Push Branch 'change/67701/4'...
Pull
Pull...
Remote
Switch To
Advanced
Merge...
Rebase...
Reset...
Create Patch...
Apply Patch...
Ignore
Show in History
Show in Repositories View
Init Git Flow
Disconnect

configure date format



Date Format

Format: User-defined format

Date format: yyyy-MM-dd HH:mm:ss

Date preview: 2016-03-02 11:07:52

Time of the commit converted to your local time zone in the format specified.

Authored Date	Committer
2016-03-02 23:04:00	Thoma
2016-03-02 21:24:14	Matthi
2016-03-02 15:25:27	Andre
2016-02-29 23:03:38	Matthi
2016-02-07 13:59:26	Thoma
2016-02-29 07:45:46	Thoma
2016-02-28 18:01:27	Thoma
2016-02-26 07:45:11	Matthi

```
clipse.egit.core/src/org/eclipse/egit/cor
clipse.egit.core/src/org/eclipse/egit/cor
clipse.egit.ui.test/src/org/eclipse/egit/u
clipse.egit.ui.test/src/org/eclipse/egit/u        est.java
clipse.egit.ui.test/src/org/eclipse/egit/u        poDeletionTest.java
```

Gerrit integration

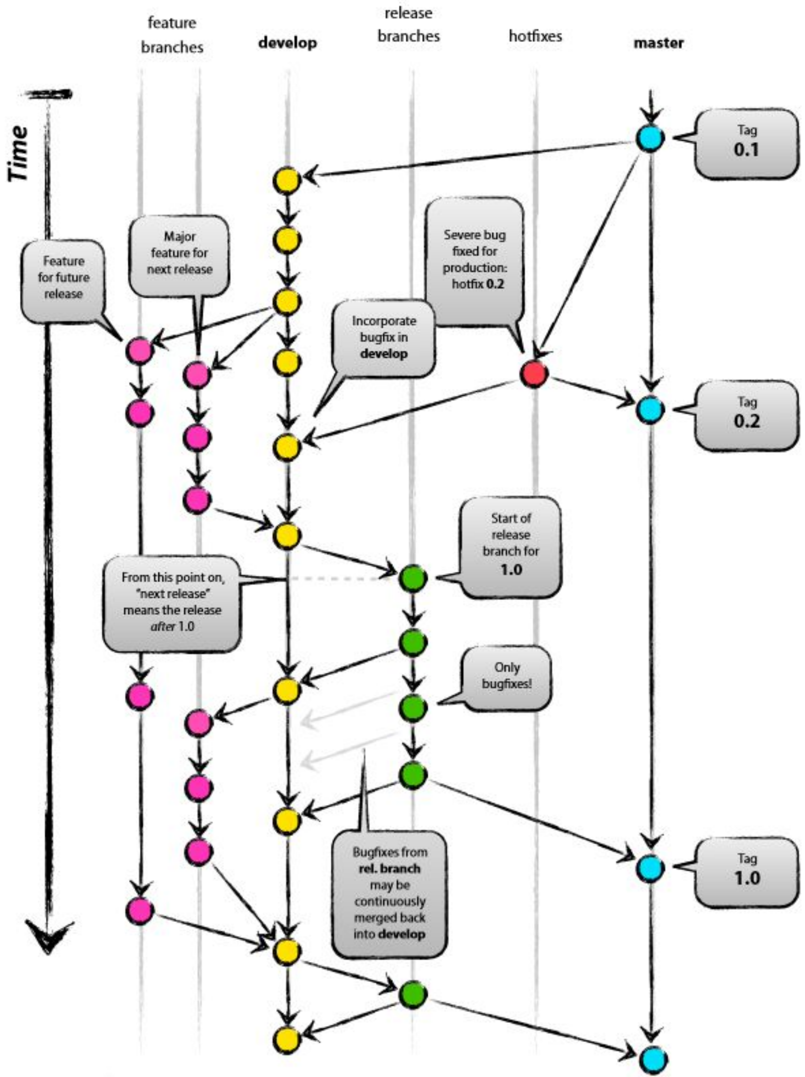
EGit 4.3

- auto-configure repository cloned from Gerrit
- improved "Fetch from Gerrit" wizard
- "Commit and push" opens "Push to Gerrit" wizard

Gitflow support

EGit

- Automation for gitflow branching strategy and workflow
- Install EGit Gitflow feature if you want to use this


feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

Gitflow in a nutshell

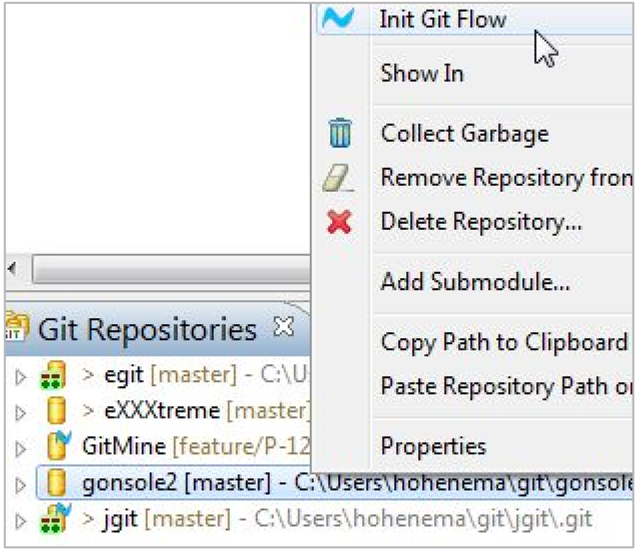
“It forms an elegant mental model that is easy to comprehend and allows team members to develop a shared understanding of the branching and releasing processes.”

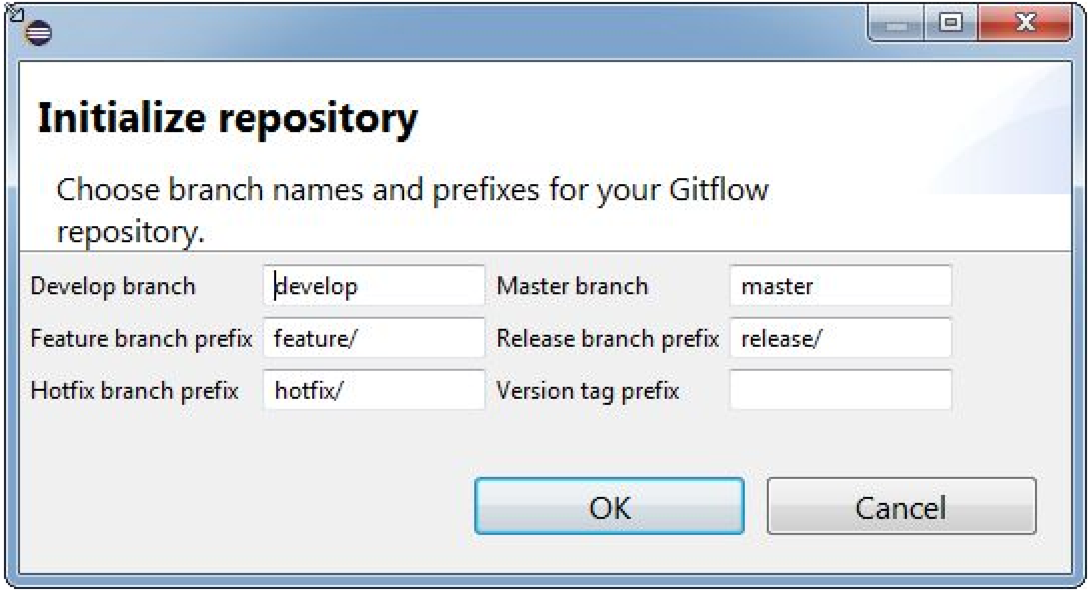
- Vincent Driessen

(creator of Gitflow)

Author: Vincent Driessen
Original blog post: http://nvie.com/posts/a-succesful-git-branching-model

Gitflow integration - initialize



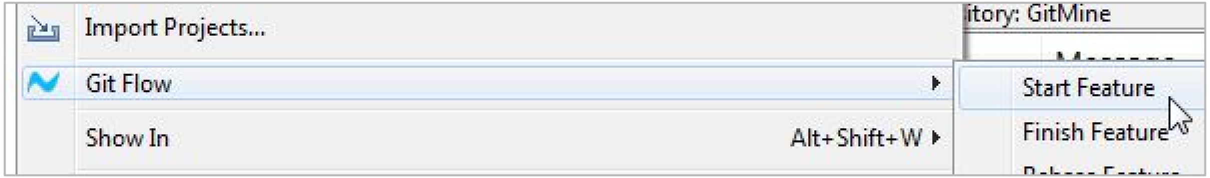


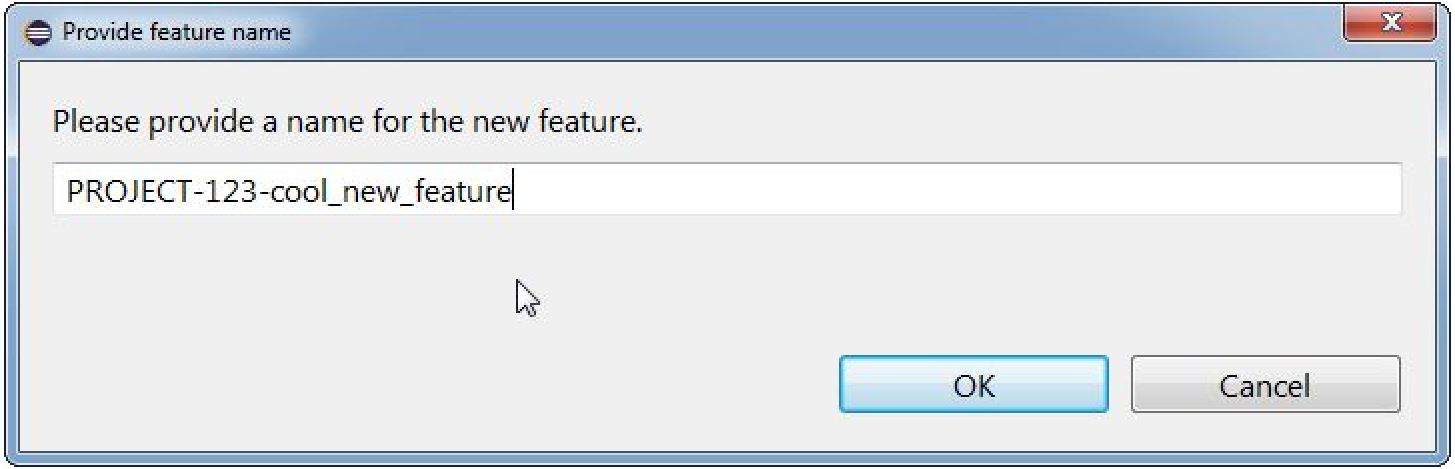
Initialize repository

Choose branch names and prefixes for your Gitflow repository.

Develop branch	develop	Master branch	master
Feature branch prefix	feature/	Release branch prefix	release/
Hotfix branch prefix	hotfix/	Version tag prefix	

Gitflow integration - create feature branch

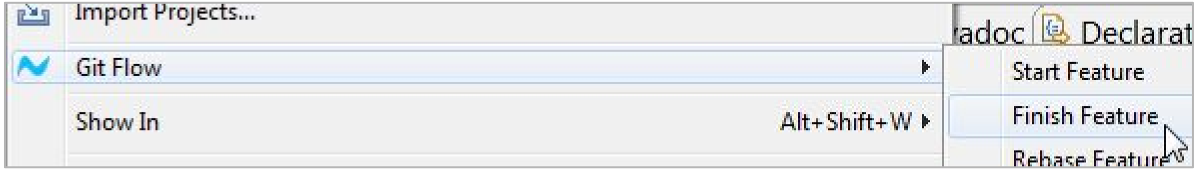


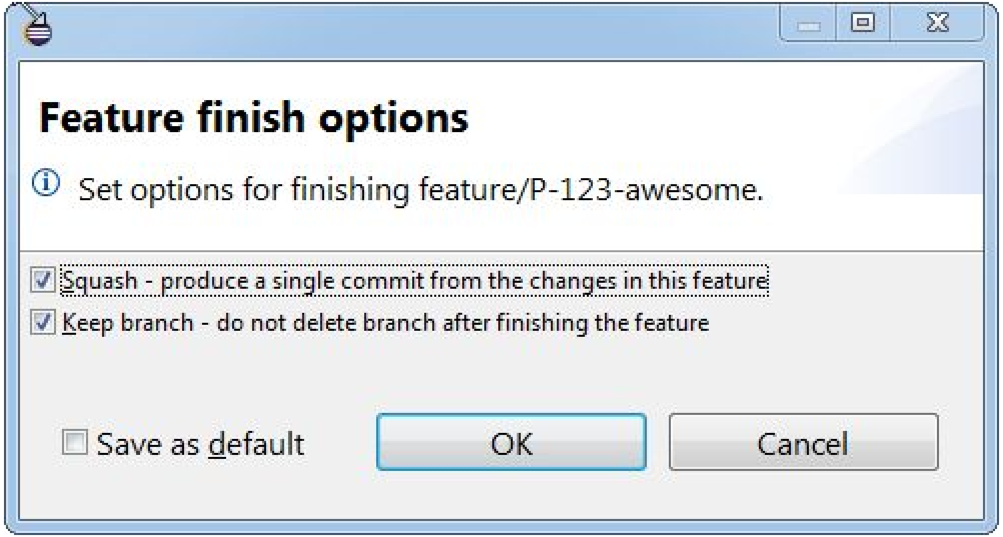


Please provide a name for the new feature.

PROJECT-123-cool_new_feature

Gitflow integration - finish feature




Feature finish options

Set options for finishing feature/P-123-awesome.

☑ Squash - produce a single commit from the changes in this feature

☑ Keep branch - do not delete branch after finishing the feature

□ Save as default

.gitattributes

JGit 4.3

fixes most wanted JGit bug (108 votes)

- filter attributes (4.2)
- text attributes
- eol attributes
- macros

LFS
Large File Storage

JGit, EGit 4.3

- integrate git-lfs extension in client (4.2)
- basic JGit LFS server (file system or S3)
- Gerrit plugins integrating JGit LFS server

Versioning large binaries in Git

large binary files can't be packed by Git efficiently

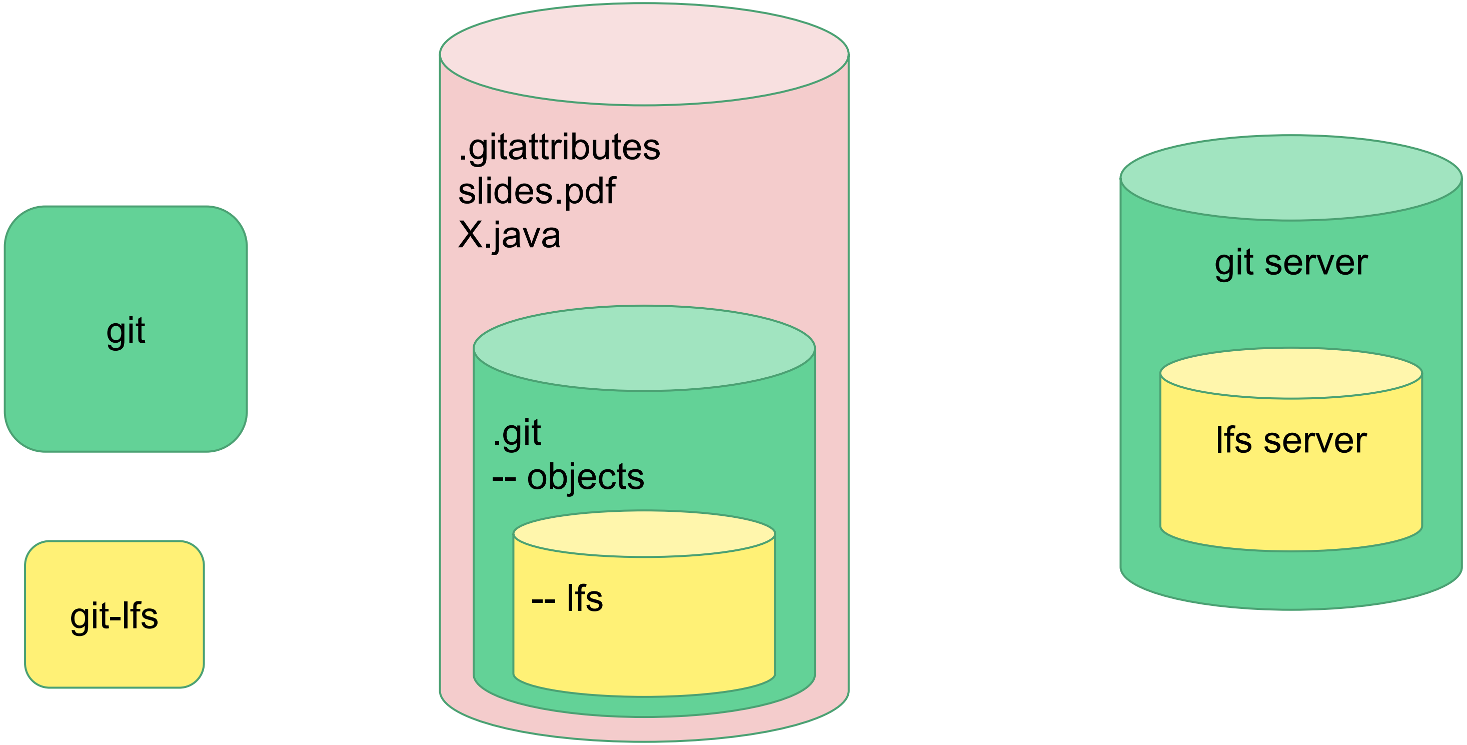
➔ Git repository growing quickly

➔ gc more expensive and less efficient

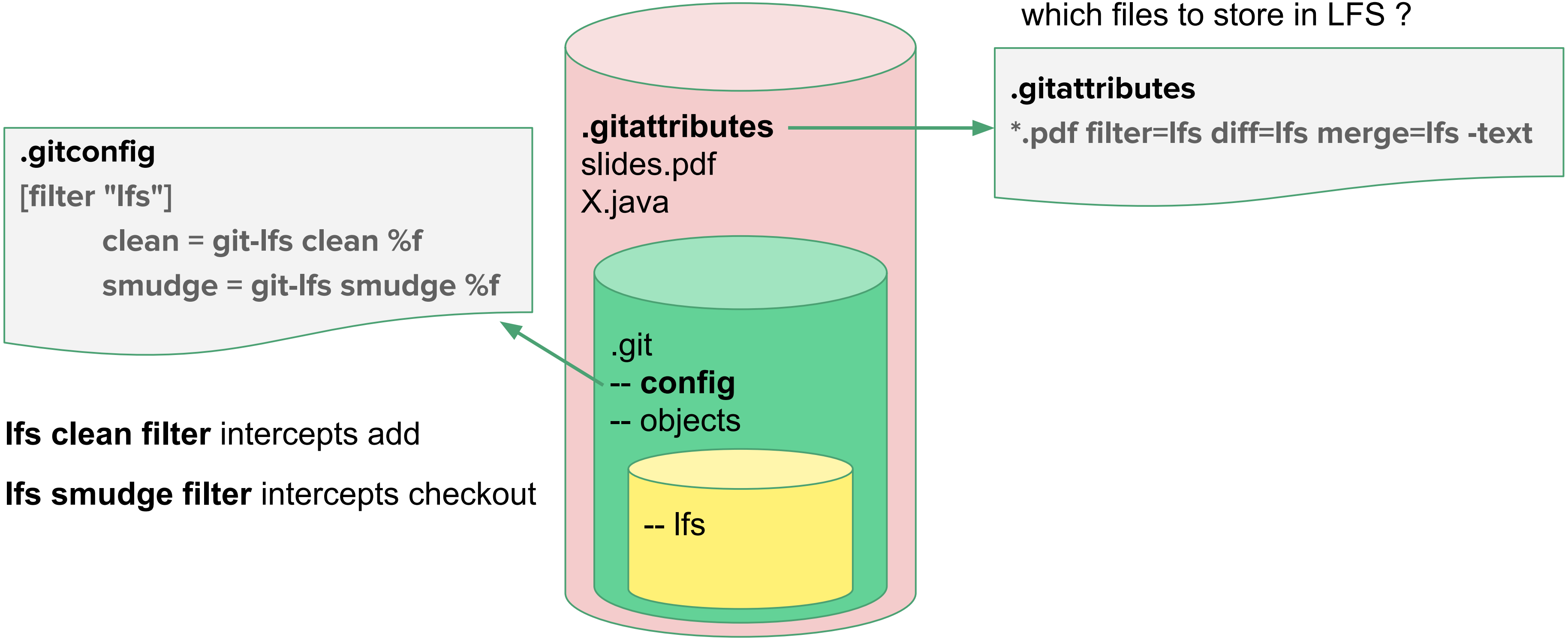
➔ slows down transport

➔ Git server load grows

Large File Storage (LFS) extension



LFS configuration



which files to store in LFS ?

.gitattributes

```
*.pdf filter=lfs diff=lfs merge=lfs -text
```

.gitattributes
slides.pdf
X.java

.git
-- **config**
-- objects

-- lfs

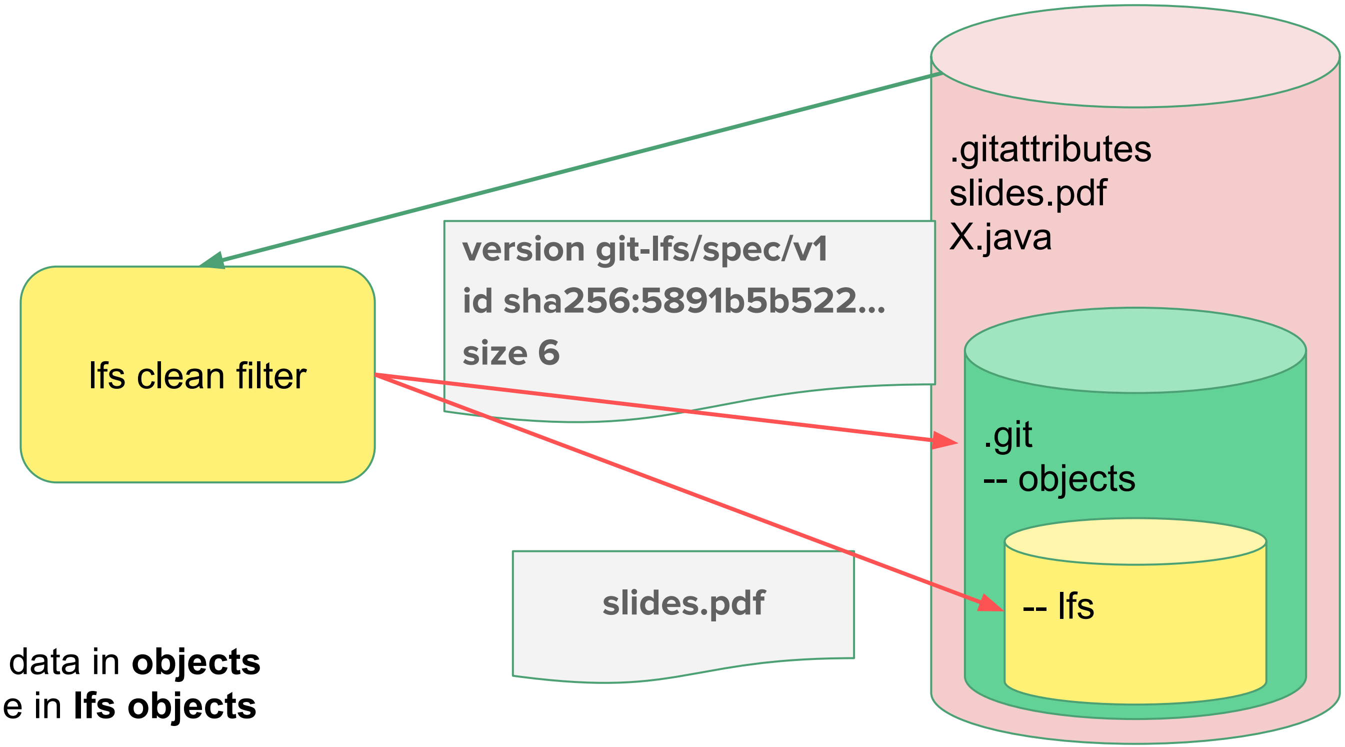
.gitconfig

```
[filter "lfs"]
        clean = git-lfs clean %f
        smudge = git-lfs smudge %f
```

lfs clean filter intercepts add

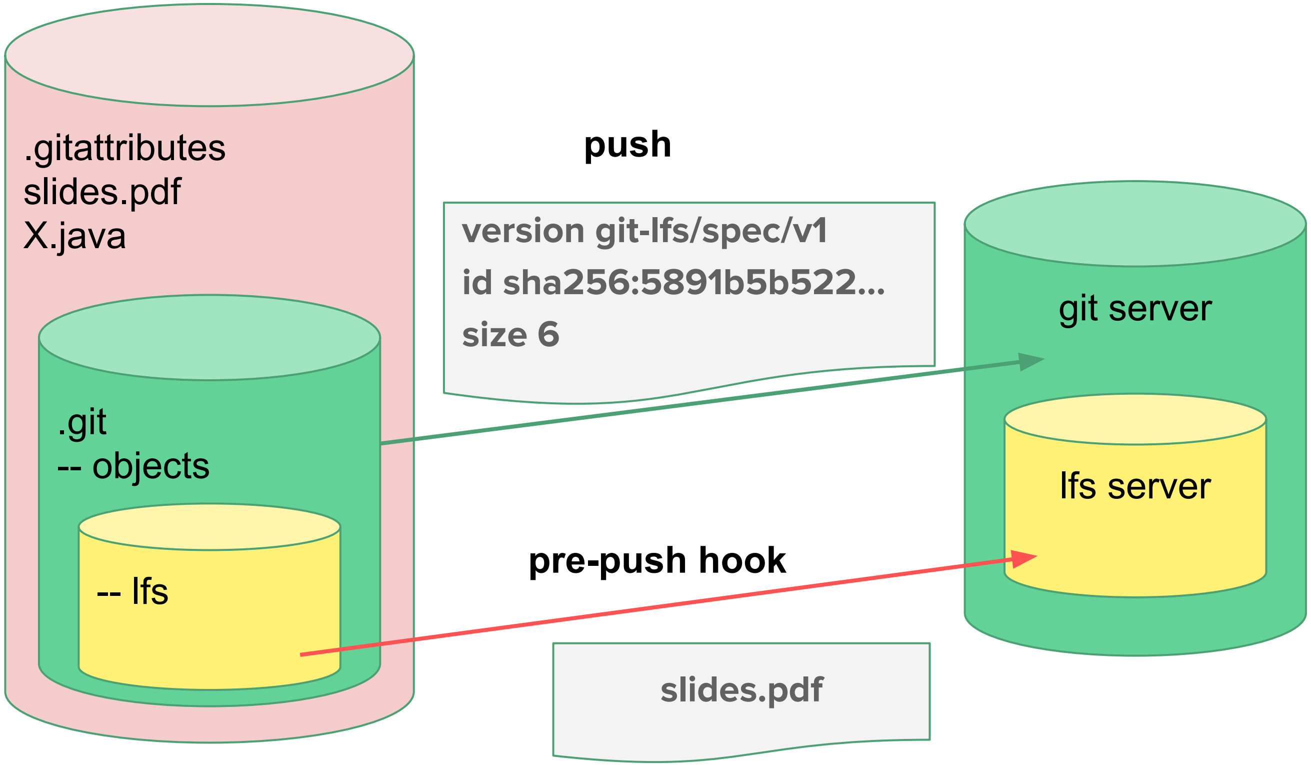
lfs smudge filter intercepts checkout

git add slides.pdf



store meta data in **objects**
store big file in **lfs objects**

git push origin



.gitattributes
slides.pdf
X.java

.git
-- objects

-- lfs

push

```
version git-lfs/spec/v1
id sha256:5891b5b522...
size 6
```

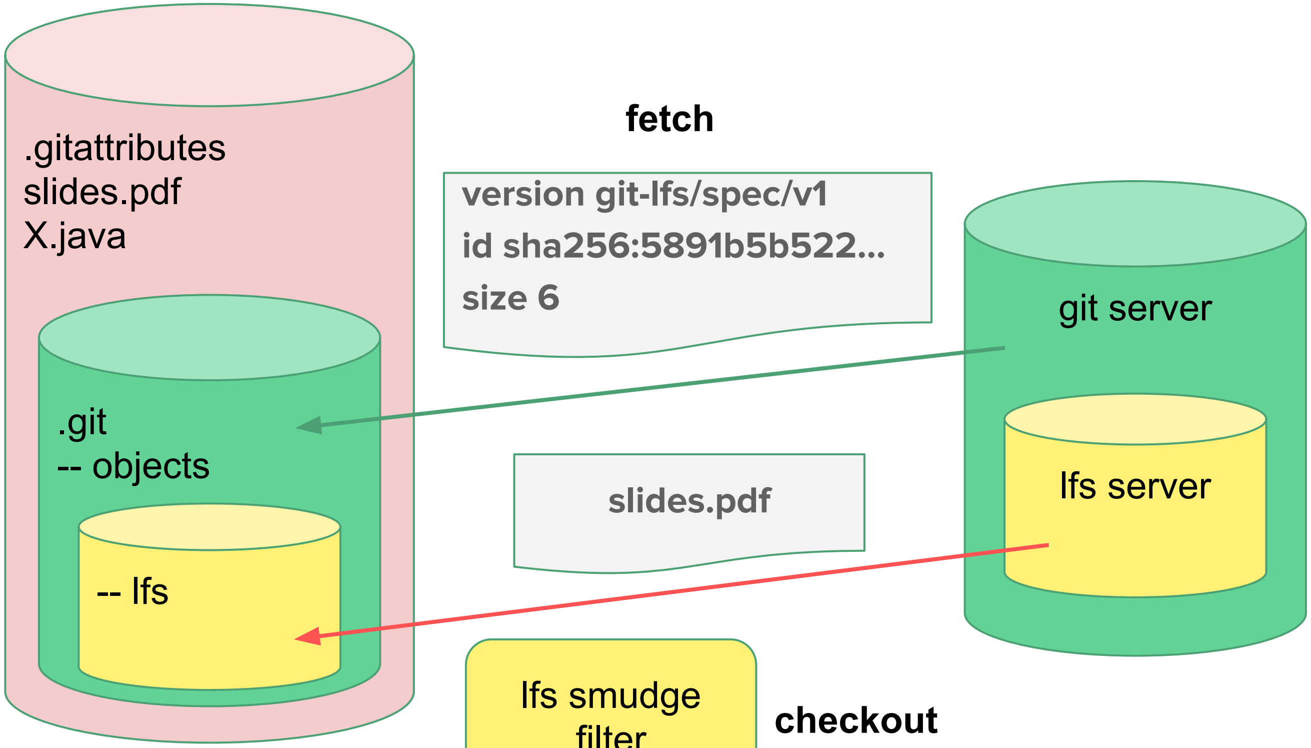
git server

lfs server

pre-push hook

slides.pdf

git fetch/clone, git checkout



fetch

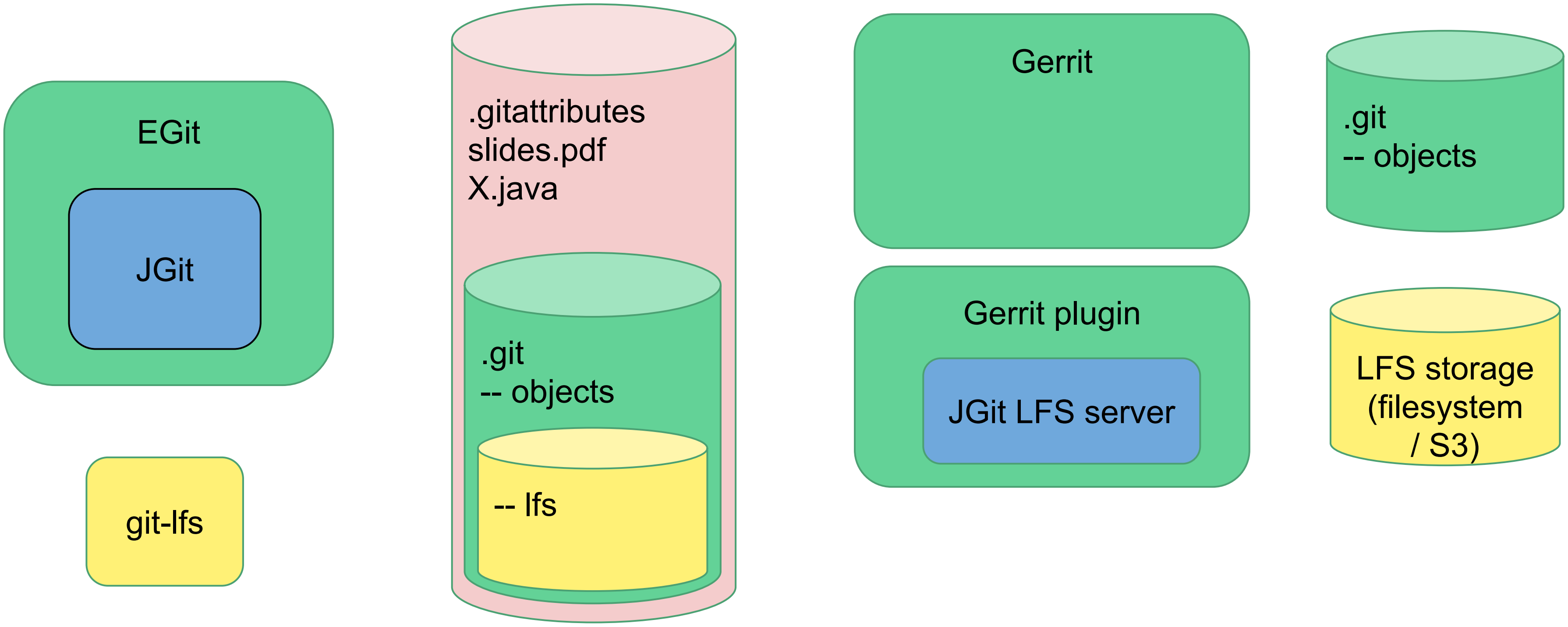
version git-lfs/spec/v1
id sha256:5891b5b522...
size 6

slides.pdf

lfs smudge filter

checkout
downloads large
object lazily

LFS server with JGit filesystem/S3 storage



➡ Practical workflows with Git LFS

Tim Pettersen (Atlassian)
Lake Anne A
Wednesday - 10:30 to 11:05
https://www.eclipsecon.org/na2016/session/practical-workflows-git-lfs

The quest for distributed JGit

JGit

RefTree (4.2)

- enable atomic push
- version refs as git objects
- push = commit containing ref updates

Ketch

- enable distributed JGit server
- Raft distributed consensus
- distributed transaction log based on RefTree

RefTree: Versioning branches in git

store refs in git tree objects

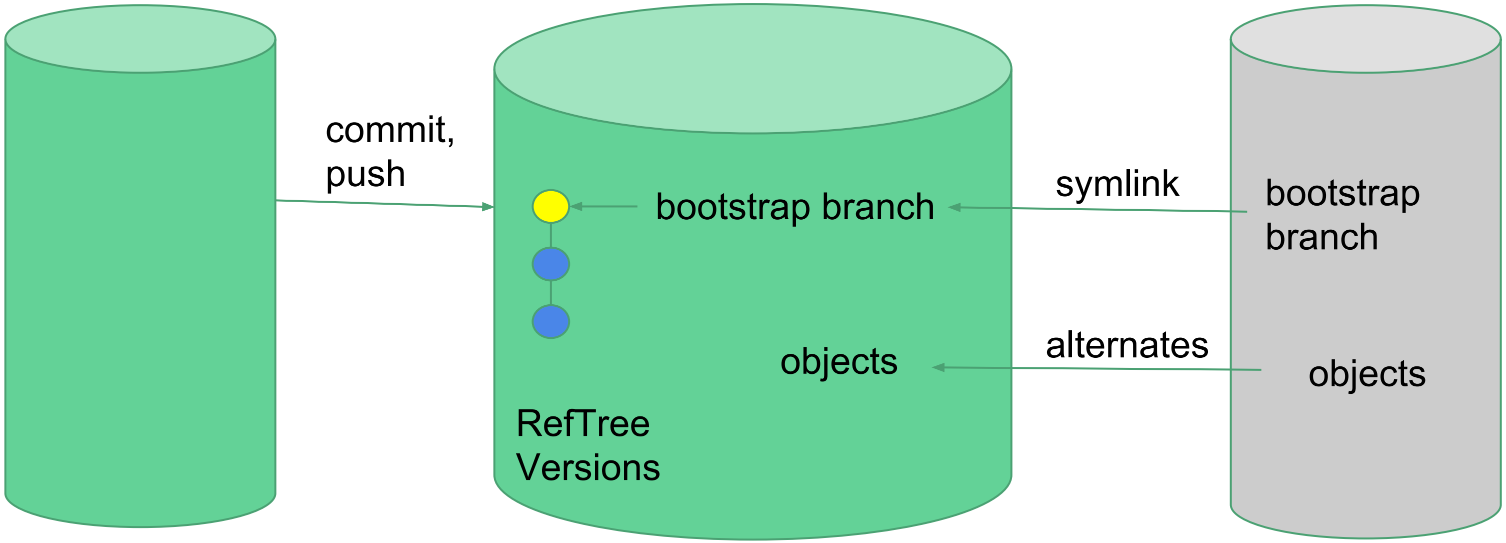
push can update many refs -> stored in one RefTree commit

-> enable transactional ref updates when receiving a pack
-> compare single SHA1 to compare repository states

jgit
clone with worktree
standard refs

jgit.git
bare repository
refs stored in RefTree

spy
empty, used for
introspection



commit,
push

bootstrap branch

objects

RefTree
Versions

symlink

alternates

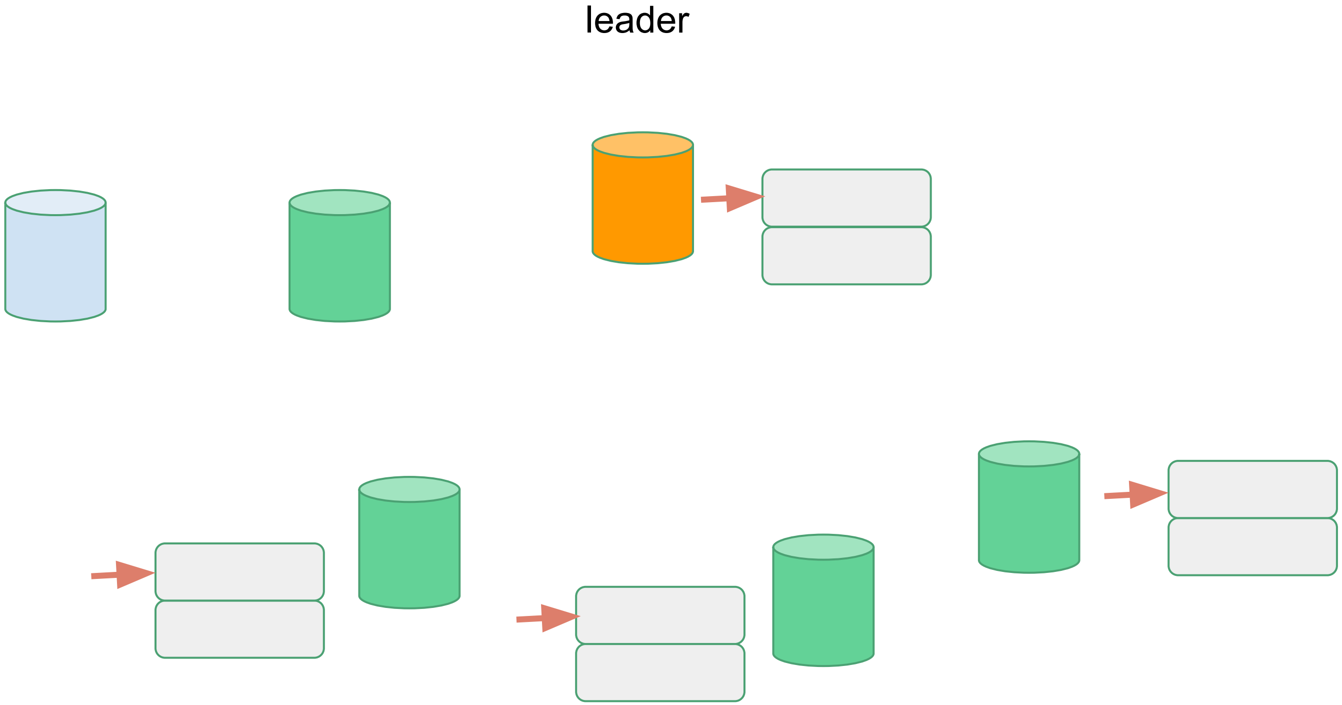
bootstrap
branch

objects

bootstrap branch symlinked
objects linked via alternates

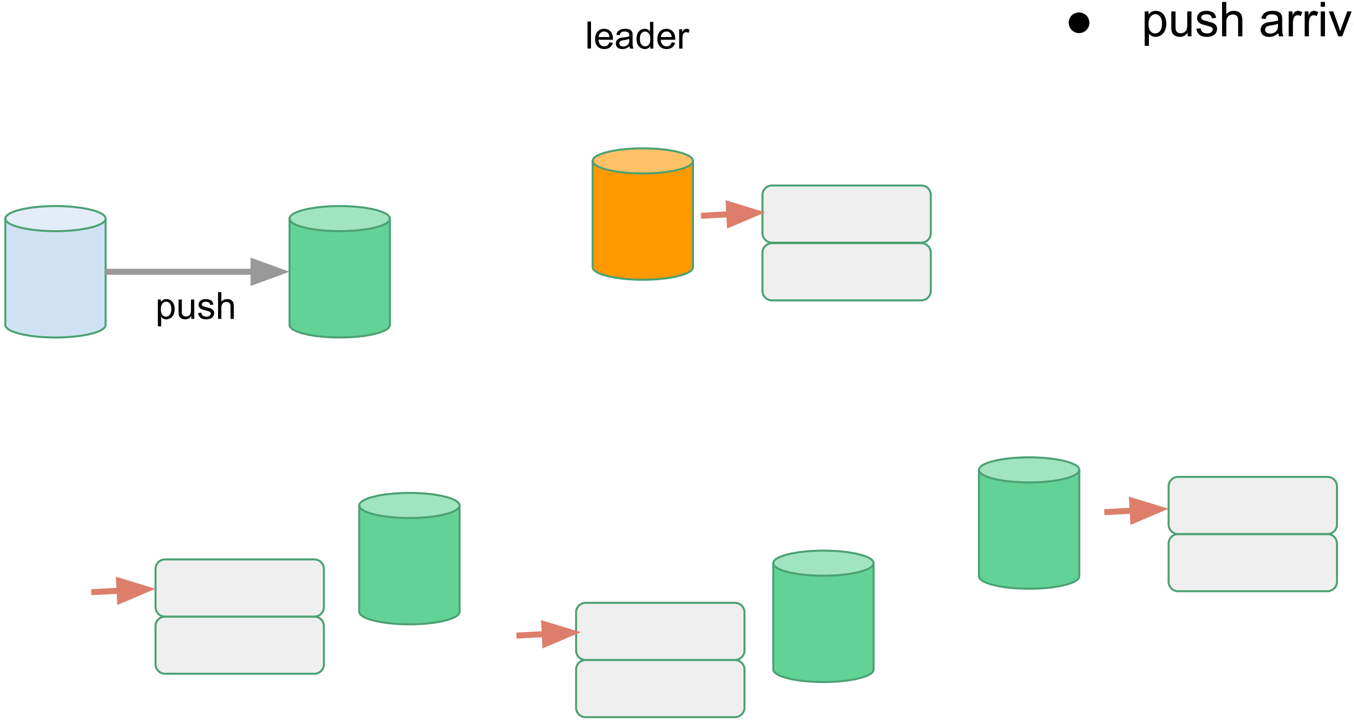
Ketch: multi-master git repository

- leader election (Raft)

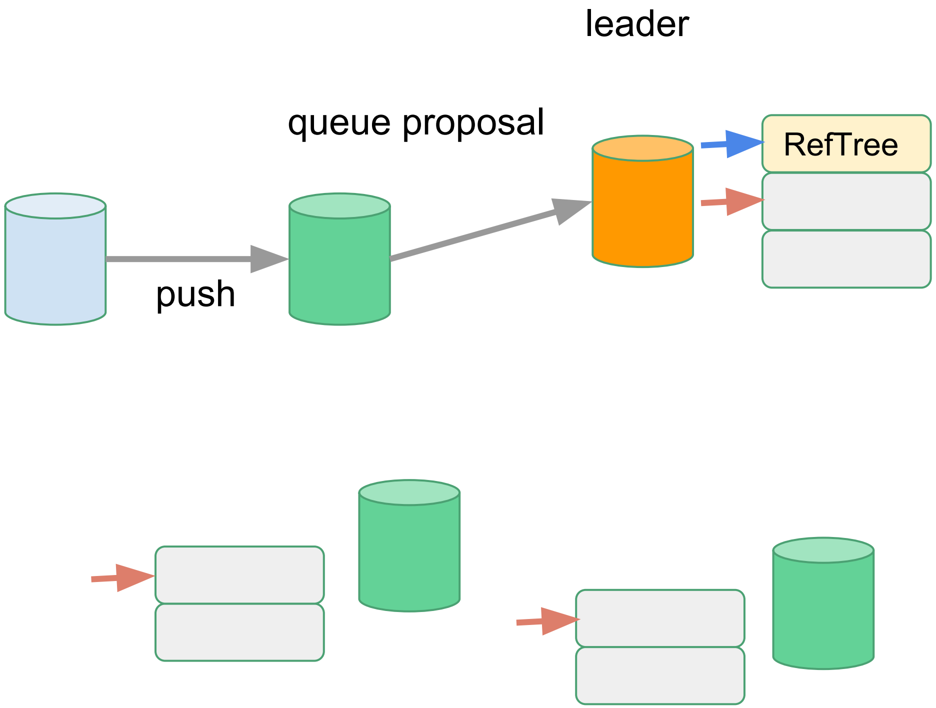


Ketch: multi-master git repository

- leader election (Raft)
- push arrives on any server

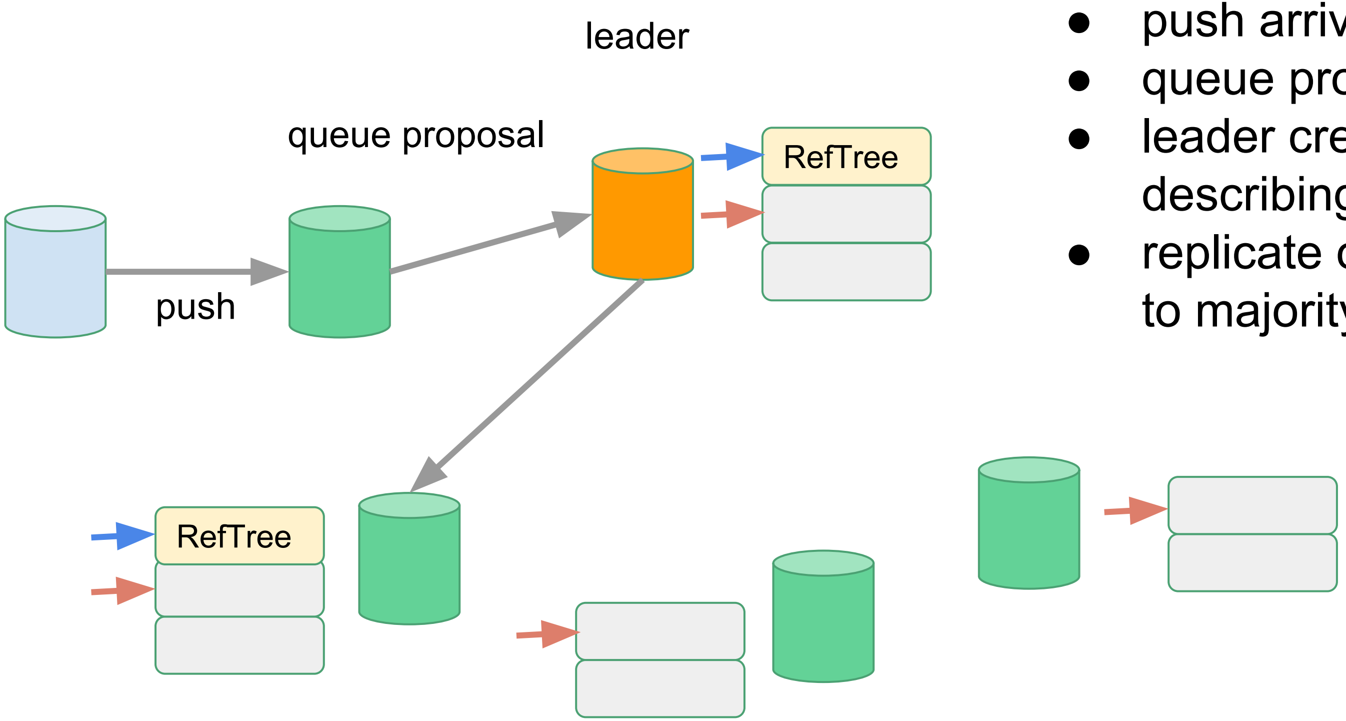


Ketch: multi-master git repository



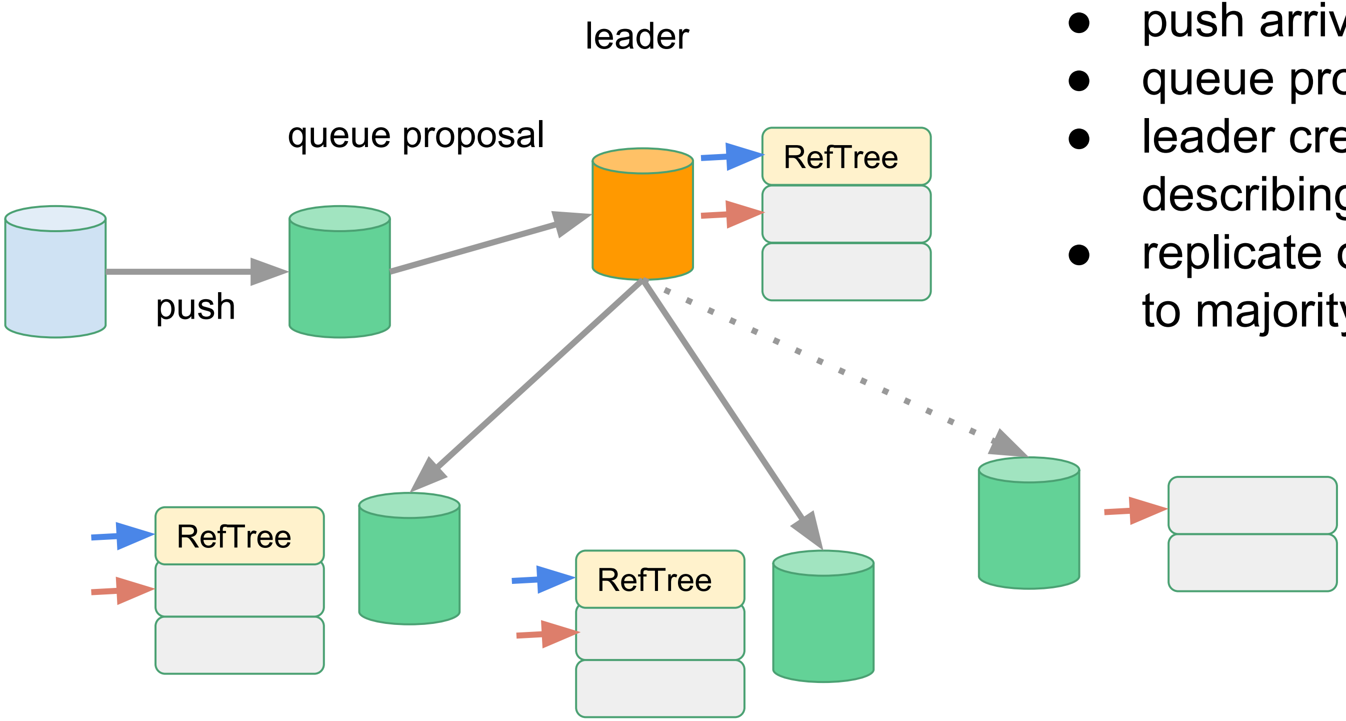
- leader election (Raft)
- push arrives on any server
- queue proposal to leader
- leader creates new RefTree describing target state

Ketch: multi-master git repository



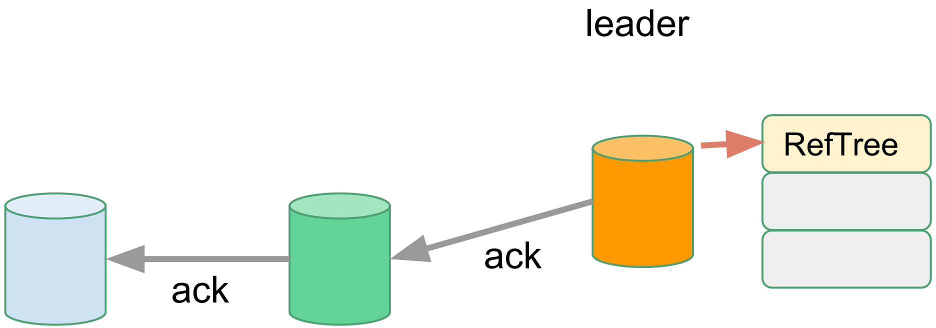
- leader election (Raft)
- push arrives on any server
- queue proposal to leader
- leader creates new RefTree describing target state
- replicate objects and RefTree to majority of servers

Ketch: multi-master git repository

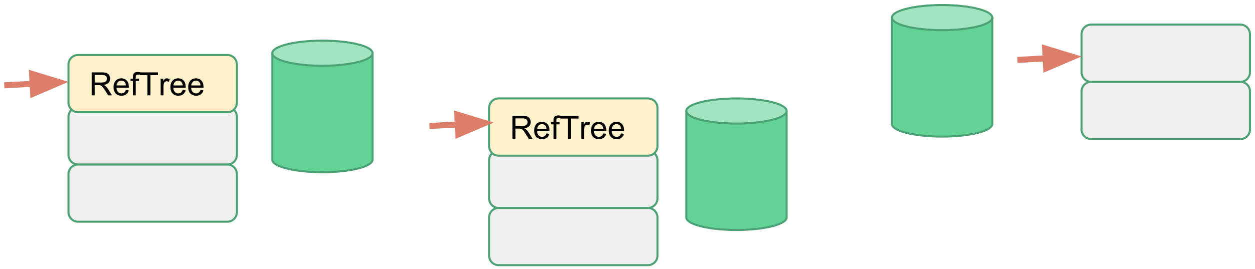


- leader election (Raft)
- push arrives on any server
- queue proposal to leader
- leader creates new RefTree describing target state
- replicate objects and RefTree to majority of servers

Ketch: multi-master git repository



- leader election (Raft)
- push arrives on any server
- queue proposal to leader
- leader creates new RefTree describing target state
- replicate objects and RefTree to majority of servers
- commit transaction



Questions & Answers

Evaluate the Sessions

Sign in and vote at **eclipsecon.org**

- 1 0 + 1