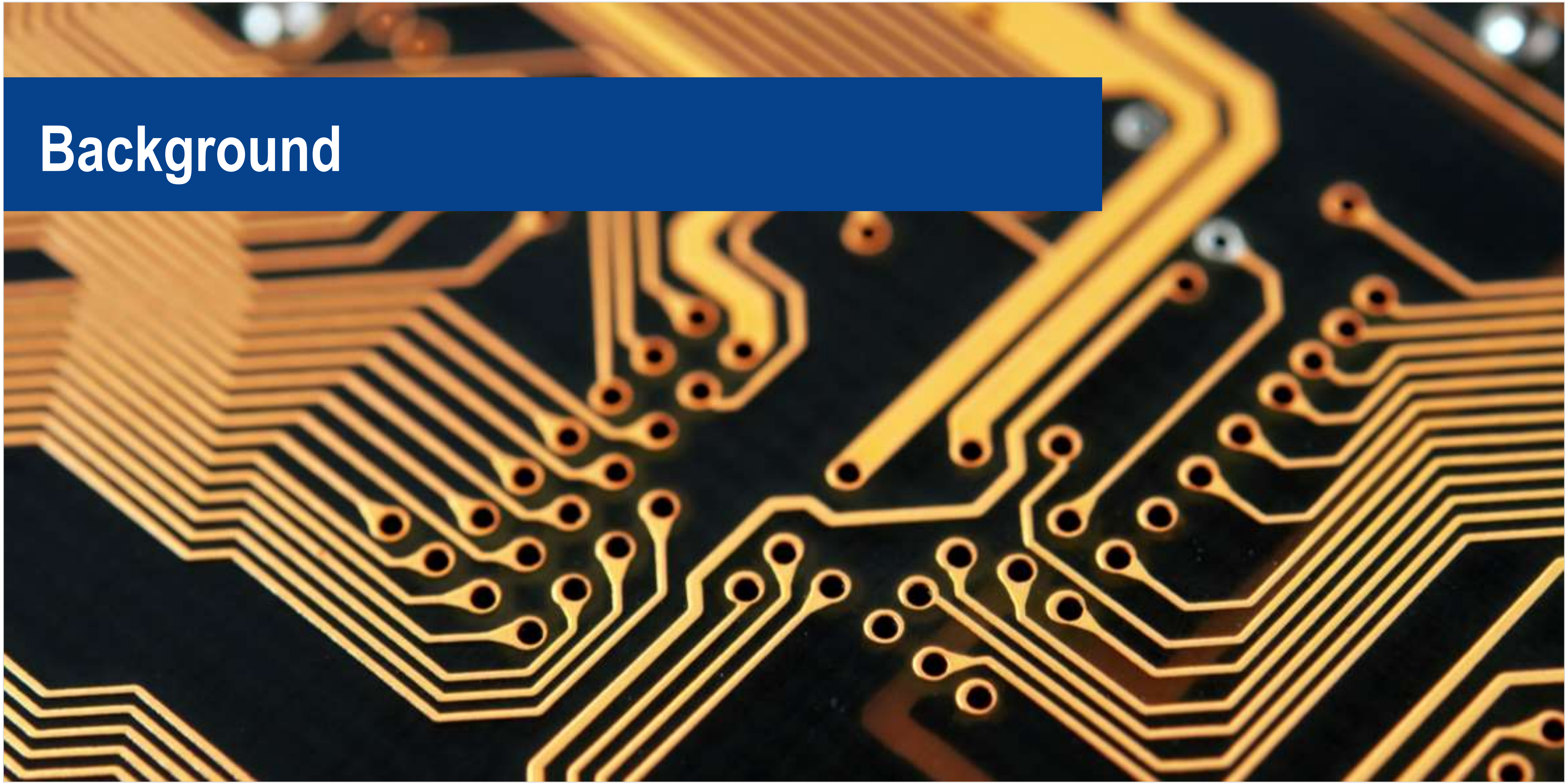
Tools Training
Git & Gerrit

William Riley
REE

Topics

- Background
  - What are Git & Gerrit?
  - Why are we using them?
- Tool Setup
  - 1st time Gerrit login
  - Setting up Tools
- Basic Operation
  - How to commit a change
  - How to review a change
- Advanced Usage
  - Drafts
  - Fixing Common Problems
  - Advanced Workflows



Background

Git Basics

- Will only provide summary of features and differences today
- Recommend you look at the following resources:
  - Pro Git http://git-scm.com/book/en/v2
  - http://www.vogella.com/tutorials/Git/article.html

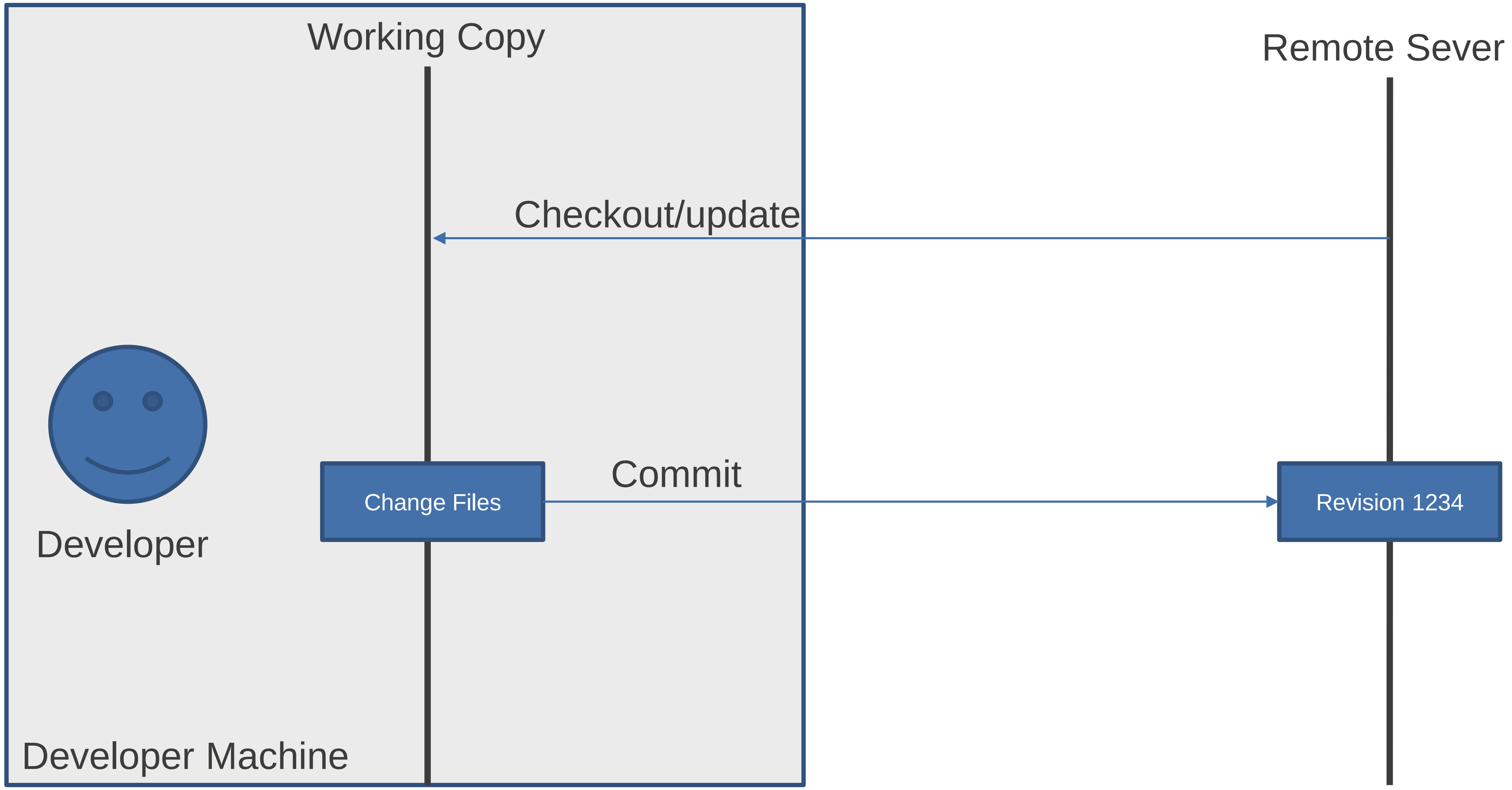
Ignore SVN

- Ignore everything you know about SVN when working with Git
- Git is fundamentally different

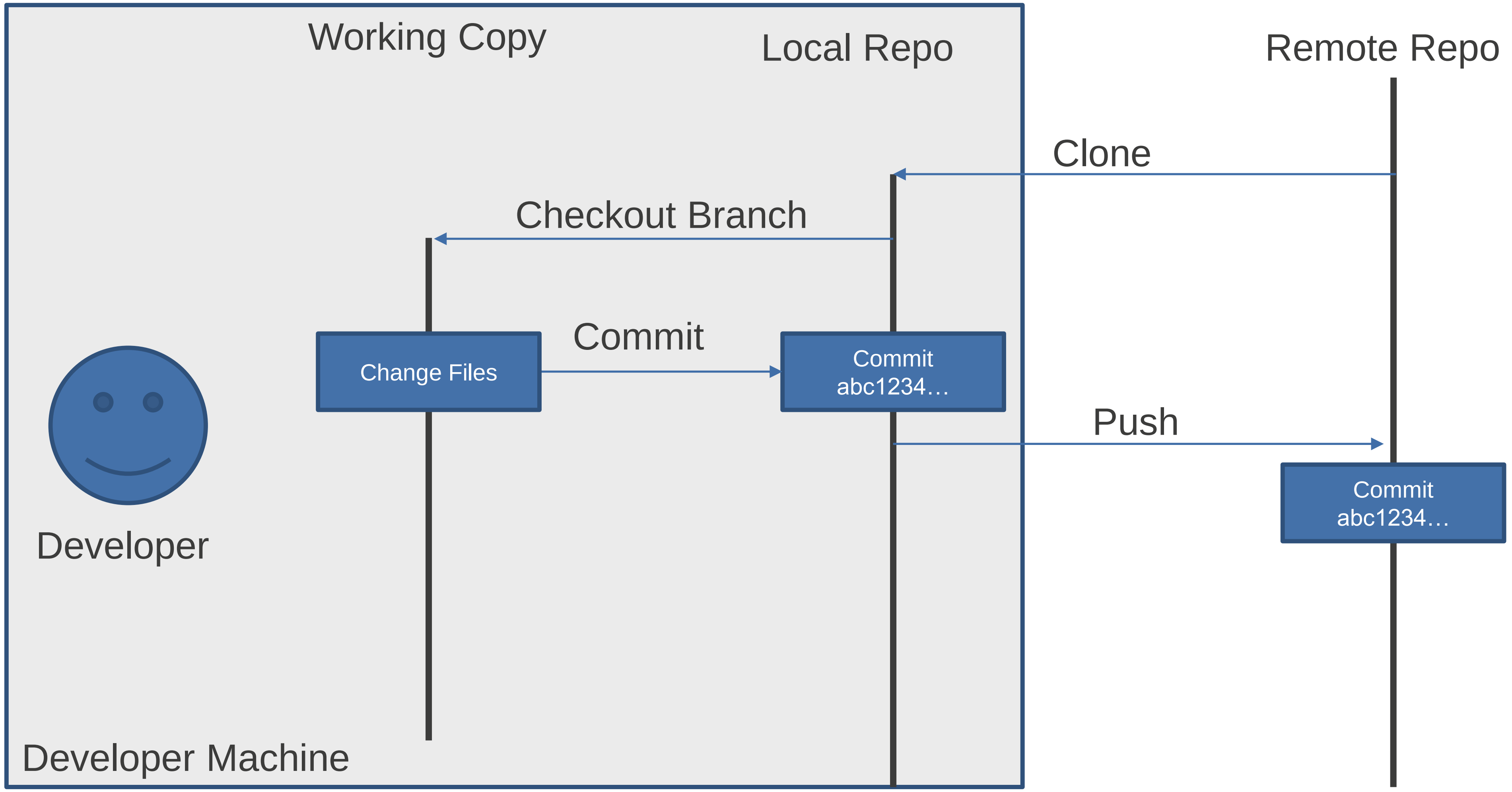
What is Git

- Git is a Distributed Version Control System (DVCS)
  - Every user has a complete copy of the repository stored locally
    - Git repositories are much smaller than SVN in most cases
    - e.g GDB Server git repo is 35MB with history back to 4.0. The source code is 288MB for 1 revision.
  - Access to file history extremely fast
  - Full functionality when disconnected from the network
  - Users can push changes directly to each other without a central server
- Originates from the Linux kernel development and was founded in 2005 by Linus Torvalds.
- Widely used by open-source projects (e.g. Eclipse) and companies (even Microsoft)

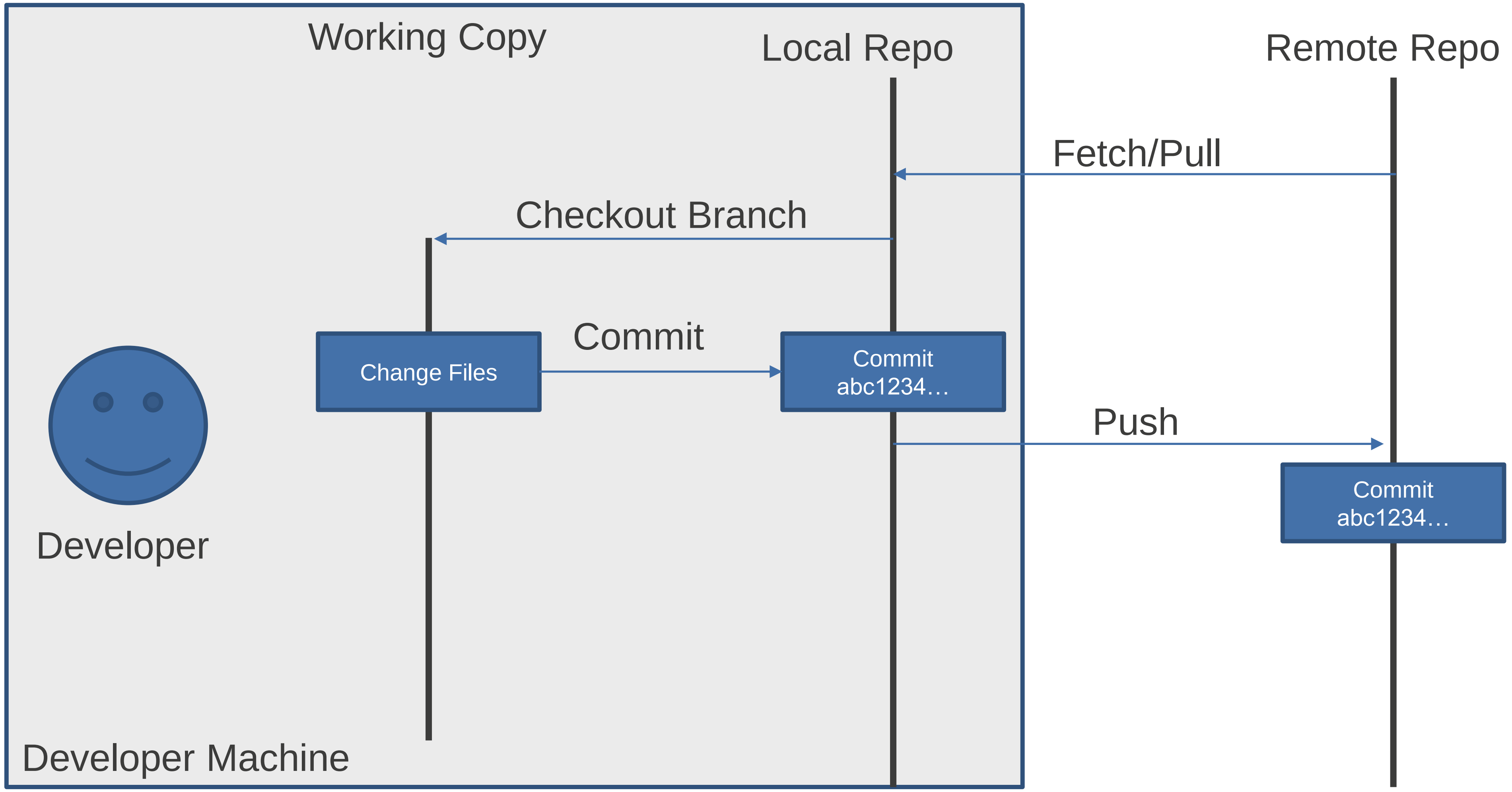
SVN Workflow



Simplified Git Workflow



Simplified Git Workflow

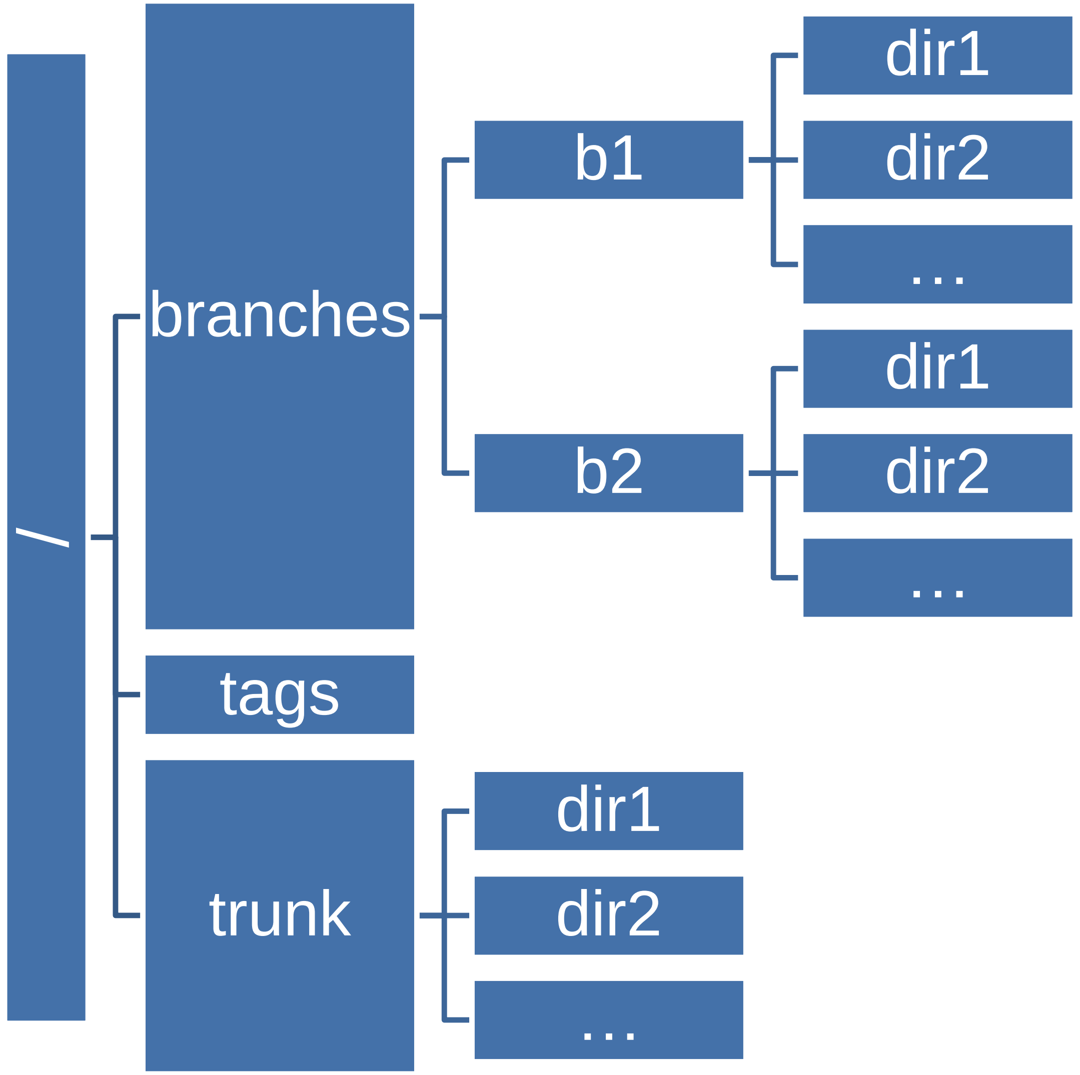


Git Terminology

- Basic Git operations
  - Clone – Creates a local copy of a remote repository. Equivalent to a checkout in SVN.
  - Fetch – Gets all branches & tags (“refs”) from a remote repository along with their contents but does not merge into your local copy
  - Pull – Same as fetch but updates local copy of the branch. Equivalent to an SVN checkout.
  - Push – Pushes a local commits/branch/tag to a remote repository.

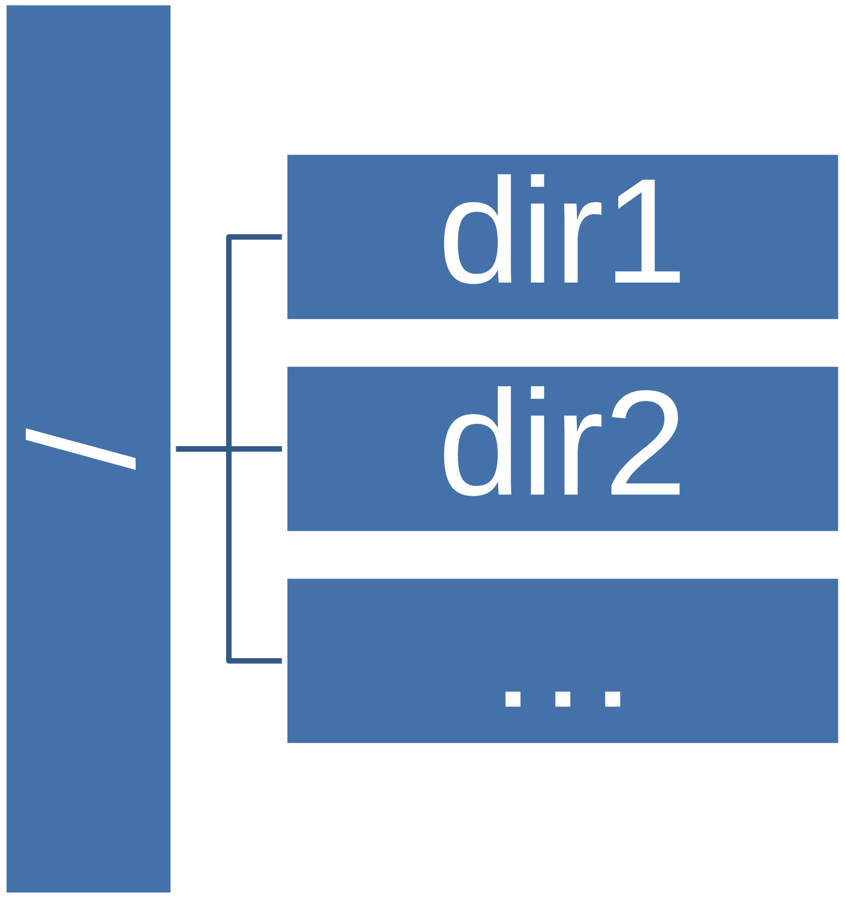
SVN Repository Layout

- SVN repositories are a single root directory
- Branches & Tags are copies of subdirectories at specific revisions
- You can checkout a specific subdirectory (e.g. just dir2 from trunk)



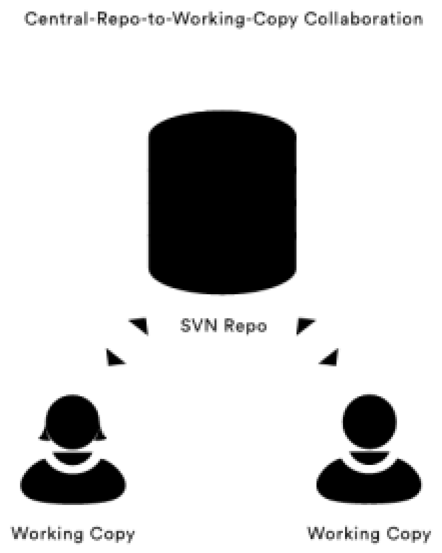
Git Repository Layout

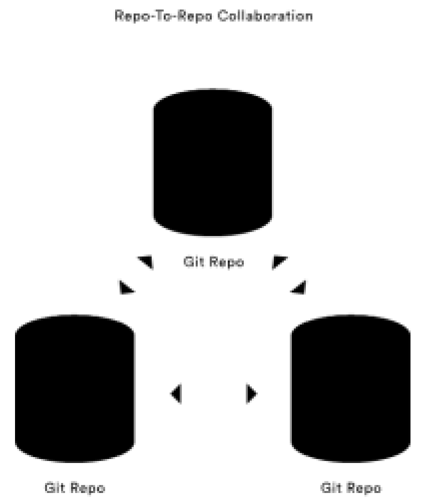
- Git performs all operations on a single root directory
- Each commit is a snapshot is this directory
- Branches & Tags are simply pointers to specific commits
- You can only clone/checkout the root directory



Important differences from SVN

- You clone the full history not just a single revision
  - This is really important to understand, everything you commit will always be downloaded by all users even if deleted later
- You clone the full tree not just individual sections
- All operations are local unless pushed to a remote server
  - Branches
  - Commits
  - Tags
- Does not handle large binary files well
  - Stores 1 complete copy per commit in some cases





What we will lose

- Closing tickets on commit
- Updating file headers with latest committer & revision (only used by GDB Server)
- Ability to restrict access by directory
- Single repository
  - Rather than 1 large repository we are splitting e2 studio into multiple repositories
    - GDB Server
    - Eclipse Plugins (some sections will be split into their own repositories)
    - Support files (1 repository per family)
    - Test Scripts* (Scripts only no binaries)
    - Multiple repositories for Additional Components
  - “docs_shared”

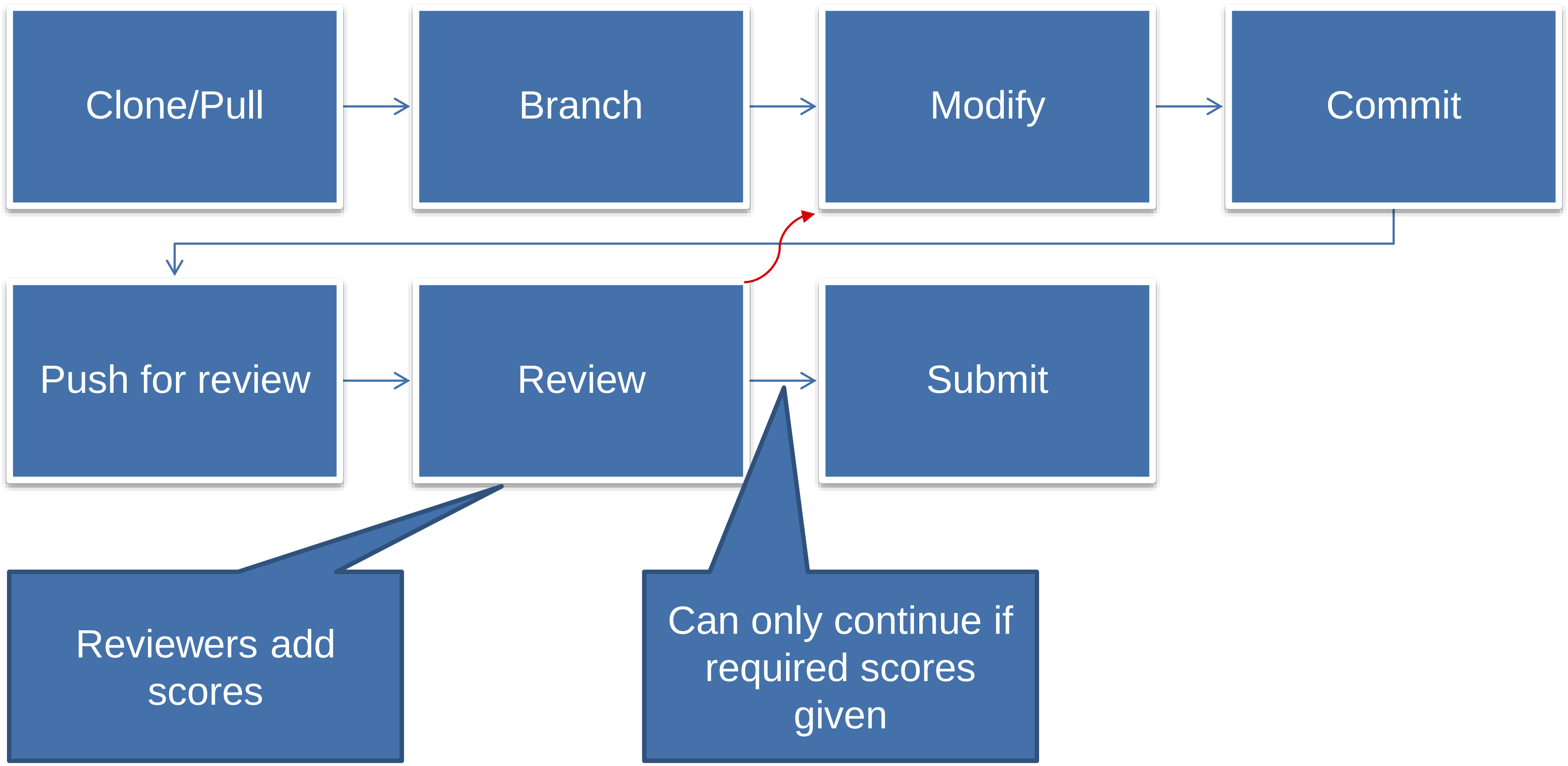
Why are we switching to Git?

- We are already using it for some parts
  - Components from open source projects we modify (e.g. CDT, Installer) are in Git repositories already
  - Single System
- Better integration with tools
  - For Eclipse development Git tools are better integrated than SVN
- Faster
- Gerrit

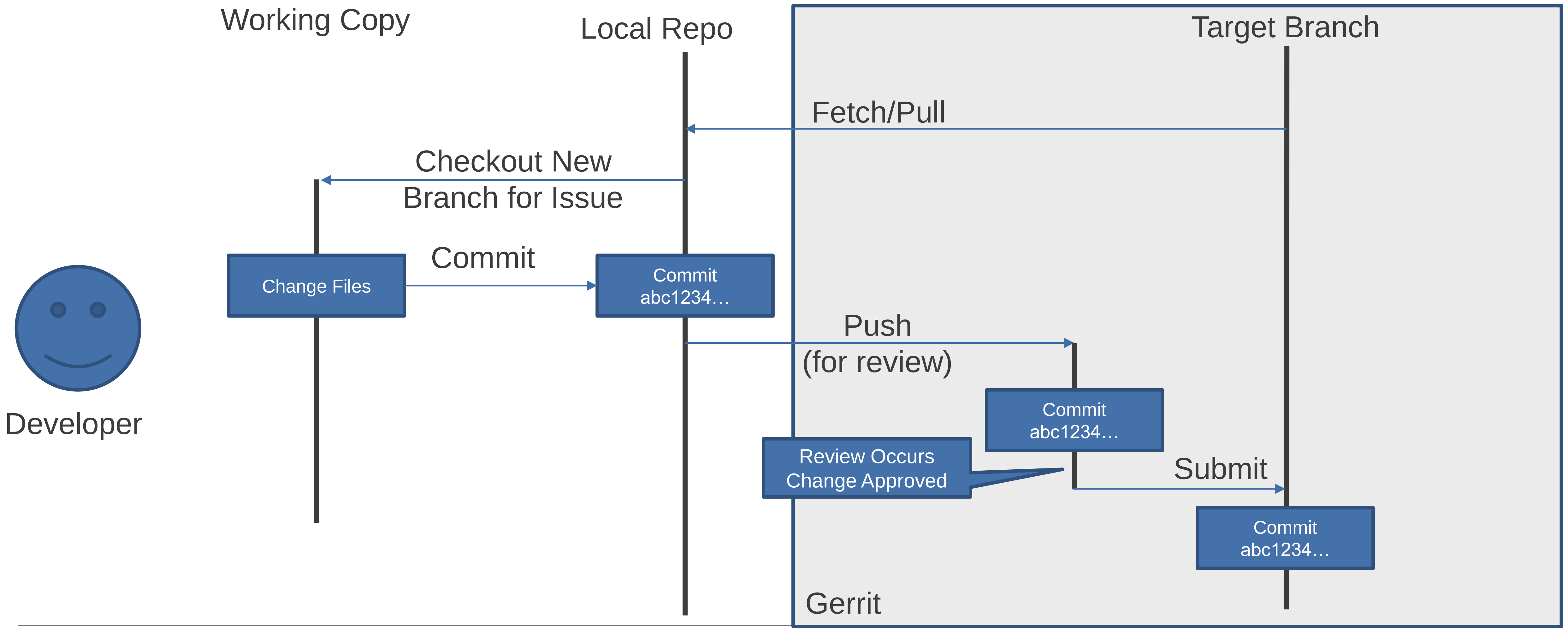
What is Gerrit?

- Gerrit is a web-based code review tool built on top of the git version control system
- Provides framework for reviewing every commit before its accepted
- Acts at a gatekeeper to the Git repository
  - Changes are pushed to Gerrit but don’t actually become a part of the project until they’ve been reviewed and accepted.

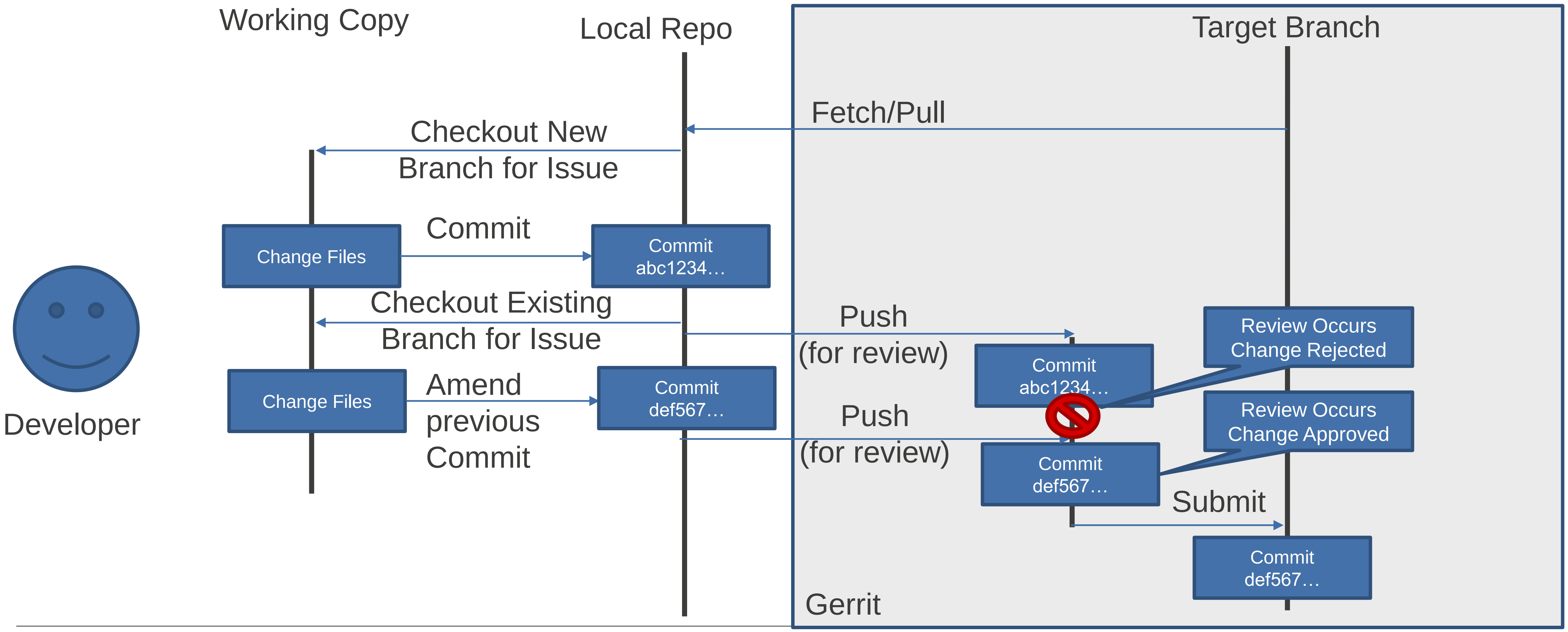
Gerrit Workflow



Simplified Gerrit Workflow



Simplified Gerrit Workflow



Why Gerrit?

- Make code review a required part of the development process
- Ensure code is reviewed before becoming part of the product
  - Prevents code review comments being forgotten
- Allows for collaborative working when changes effect plugins developed by multiple teams
- Prevents developers accidentally making changes to main Git repo which would cause other developers issues
  - You can only break your own copy!

Important rules

- Delete old code, do not just comment it out!
  - This makes reviews much easier
- Do not add redundant comments (e.g. start/end change)
- Do not duplicate files for multiple versions, overwrite!
  - (e.g. com.renesas.cdt.debug.memoryusage.help/html/e2studio401_MemoryUsage_files/image009.jpg)
  - This allows git to delta compress rather than an entirely new file
- Avoid mixing reformatting changes with other changes
  - Reviewing difficult if code has also been reformatted
- Use UTF-8 encoding for source files
  - Diff tool in Gerrit does not work with Shift-JIS or UTF-16
  - We will be converting all source files during migration

Important rules

- No files larger than 3MB*
- Gerrit will reject a push containing any larger than 5MB*

Migration

- Planning to migrate from SVN to Git in phases
  - At the start of each phase we will make the effected area in SVN read only
- Only “trunk” will be migrated, no other branches
- For GDB server & Renesas_Plugins we will be taking the history back to revision 15745 (4.0 branch -> trunk copy)

Area	Start Date	REE Internal Testing	Go Live
GDB Server	09/11	10/11	11/11
Eclipse Plugins	11/11	16/11	18/11
Support Files	09/11	13/11	17/11
Additional Components	04/11	*	*

- If servers not accessible outside REE will wait

Migration

- Those of you here already have your account details
- Everyone else will receive their details before we go live with the 1st repository

Migration (Java Developers)

- We will be modifying the structure of the “Renesas_Plugin” area
- Plugins will be moved into subdirectories to allow simpler implementation of Gerrit rules & default reviewers
- All .NET component DLL’s (e.g. com.renesas.cg/CodeGenerator) will be split into separate repositories.
  - This is due to the large size and binary nature
- All .jar files will be removed.
  - We will now be bundling libraries as separate eclipse plugins so they can be shared
  - Any additional library requests should be directed to REE

Setting up your tools

What tools do you need?

- We will use Egit & Mylyn as the Git/Gerrit client
  - For Java developers these are included in the standard e2 studio development environment image for 5.0 (new version)
  - Everyone else, use the e2 studio development environment image
  - For this training we have a copy on a USB drive. Ask if you need it. (Password is “e2studio!”)
- Why EGit?
  - Gerrit requires a change id to be added to every commit, EGit can do this automatically
  - “Push to Gerrit” which simplifies the push operation
  - Provides a simple GUI for pulling ongoing reviews out for local testing
- Good Egit tutorial: http://www.vogella.com/tutorials/EclipseGit/article.html

What tools do you need?

- Web based tools:
  - Gerrit: http://172.29.44.217
  - Bitbucket: http://172.29.44.219:7990

All tools share your JIRA username & password

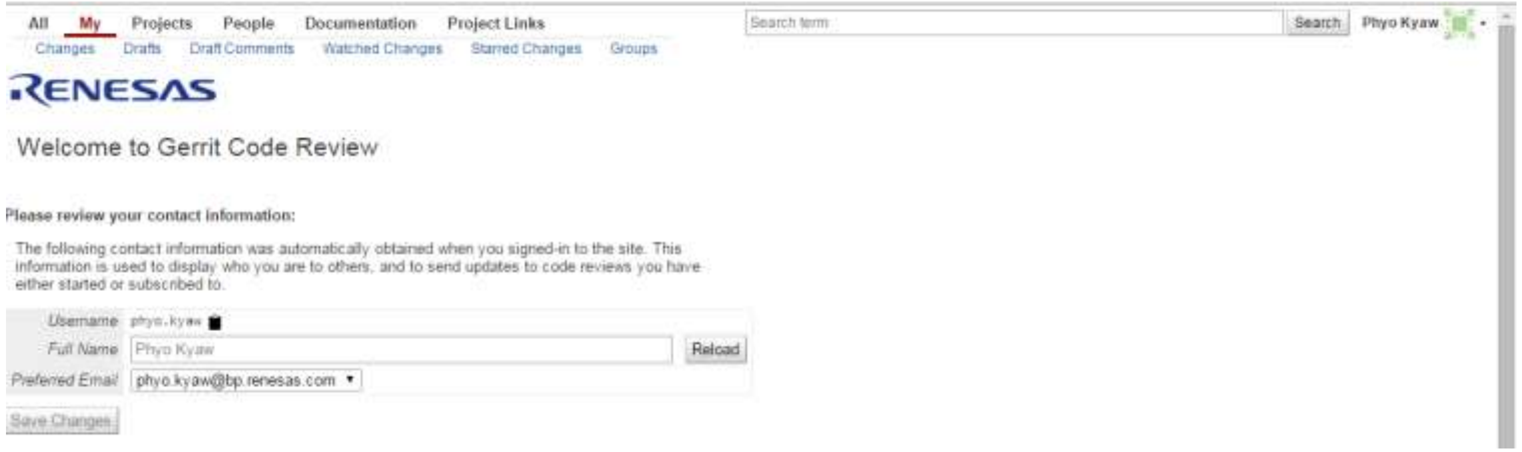
Logging into Gerrit

Before you can use Gerrit from any other tools you 1st need to log in via the Web UI

1. Open your web browser and go to http://172.29.44.217
2. When prompted enter your JIRA username and password*1
   1. If you see “FORBIDDEN” please contact William Riley (william.riley@renesas.com)

Logging into Gerrit

1. If see the following page:



Welcome to Gerrit Code Review

Please review your contact information:

The following contact information was automatically obtained when you signed-in to the site. This information is used to display who you are to others, and to send updates to code reviews you have either started or subscribed to.

Username	phyo.kyaw
Full Name	Phyo Kyaw
Preferred Email	phyo.kyaw@bp.renesas.com

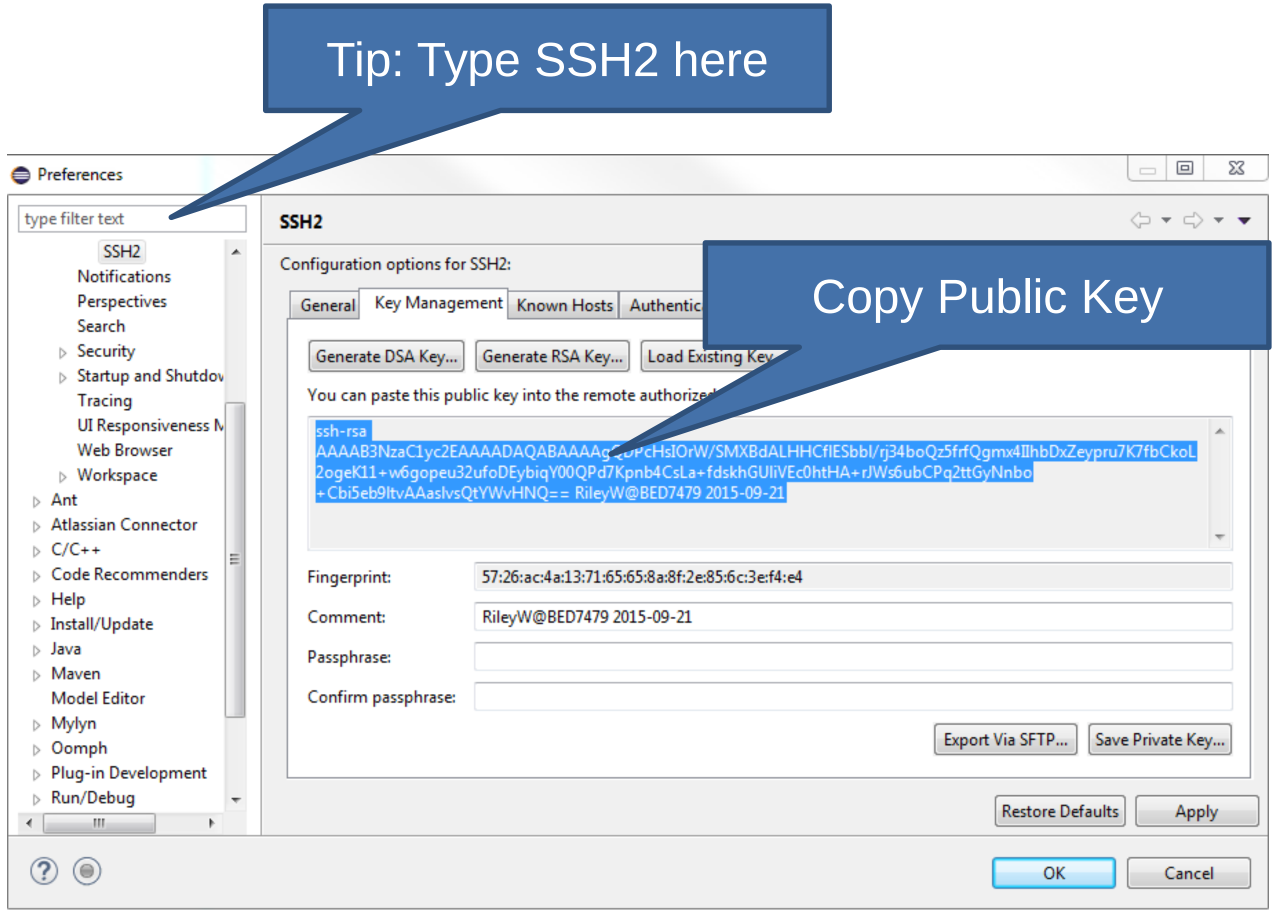
2. Scroll to the bottom and click “Continue”
3. Go to Projects -> List and check you can see all the projects you need
   1. For the training this is “Demo-Projects/eclipse-plugin-demo”
   2. If not contact William Riley (william.riley@renesas.com) to have your account added to the appropriate user groups.

Setting up Gerrit with SSH

- We recommend you use SSH to access Gerrit for all repository operations
- To do this you need to create an SSH key and add it to Gerrit

1. Generate Key

- In Eclipse go to Window - > Preferences -> Network Connections -> SSH2
- Open Key Management Tab
- Click Generate RSA Key
- Enter comment.
  - Suggest machine name & date created
- Optionally enter a passphrase
- Click Save Private key
- Suggest using default name and location
- Copy public key for next step



SSH2

Configuration options for SSH2:

You can paste this public key into the remote authorize

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAAAgQDPcHsIOrW/SMXBdALHHCflESbbl/rj34boQz5frfQgmx4IlhbDxZeypru7K7fbCkoL
2ogeK11+w6gopeu32ufoDEybiqY00QPd7Kpnb4CsLa+fdskhGUliVEc0htHA+rJWs6ubCPq2ttGyNnbo
+Cbi5eb9ltvAAaslvsQtYWvHNQ== RileyW@BED7479 2015-09-21
```

Fingerprint: 57:26:ac:4a:13:71:65:65:8a:8f:2e:85:6c:3e:f4:e4

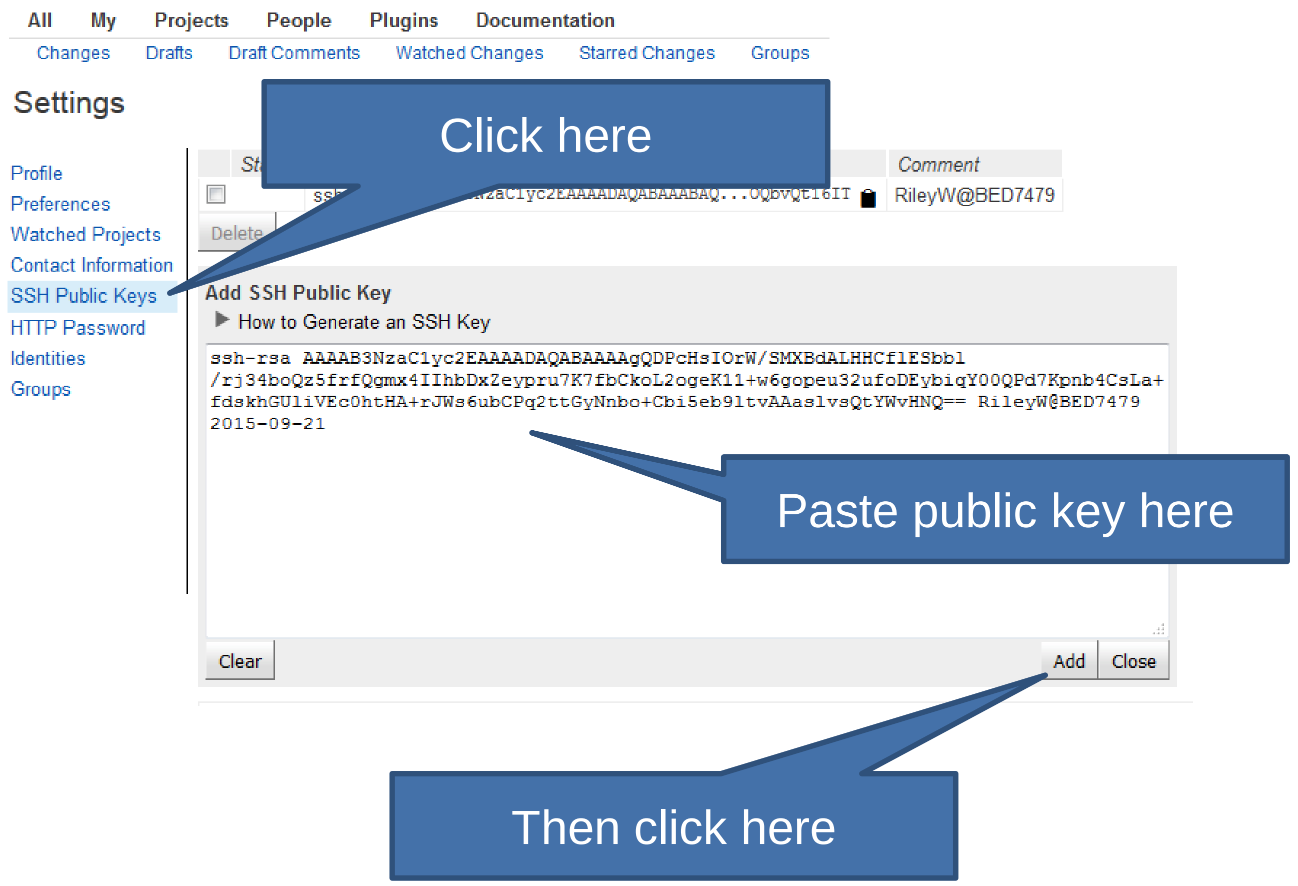
Comment: RileyW@BED7479 2015-09-21

Passphrase:

Confirm passphrase:

2. Add Key to Gerrit

- Log into Gerrit
- Click your name at top right then click settings
- Go to SSH Public Keys
- Click “Add Key”
- Paste your public key in the text box and click “Add”

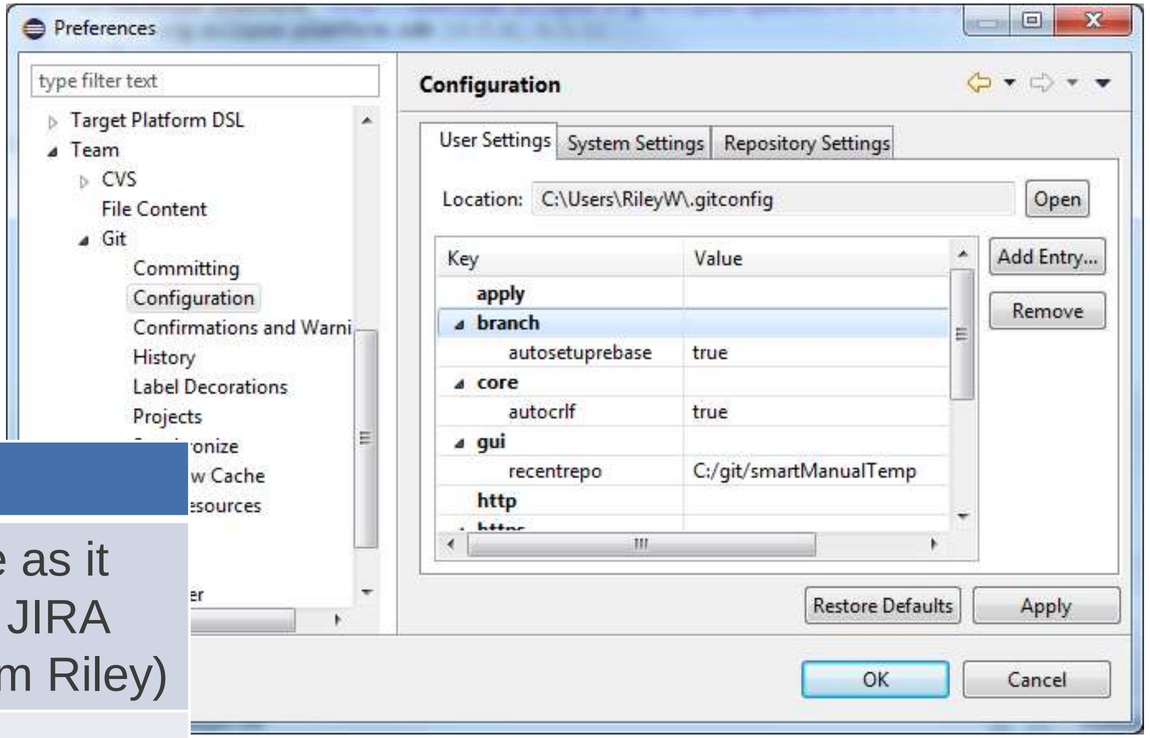


```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAAAgQDPcHsIOrW/SMXBdALHHCflESbbl
/rj34boQz5frfQgmx4IIhbDxZeypru7K7fbCkoL2ogeK11+w6gopeu32ufoDEybiqY00QPd7Kpnb4CsLa+
fdskhGUliVEc0htHA+rJWs6ubCPq2ttGyNnbo+Cbi5eb9ltvAAaslvsQtYWvHNQ== RileyW@BED7479
2015-09-21
```

3. Set up Git Config

- In Eclipse Preferences open “Team -> Git -> Configuration”
- Select “User Settings”
- Add following entries to “User Settings”:

Key	Value
user.name	Your Name as it appears in JIRA (e.g. William Riley)
user.email	Your email address
core.autocrlf	true
branch.autosetuprebase	true



Preferences

type filter text

- Target Platform DSL
- Team
  - CVS
  - File Content
  - Git
    - Committing
    - Configuration
    - Confirmations and Warni
    - History
    - Label Decorations
    - Projects
    - onize
    - w Cache
    - esources
    - er

Configuration

User Settings | System Settings | Repository Settings

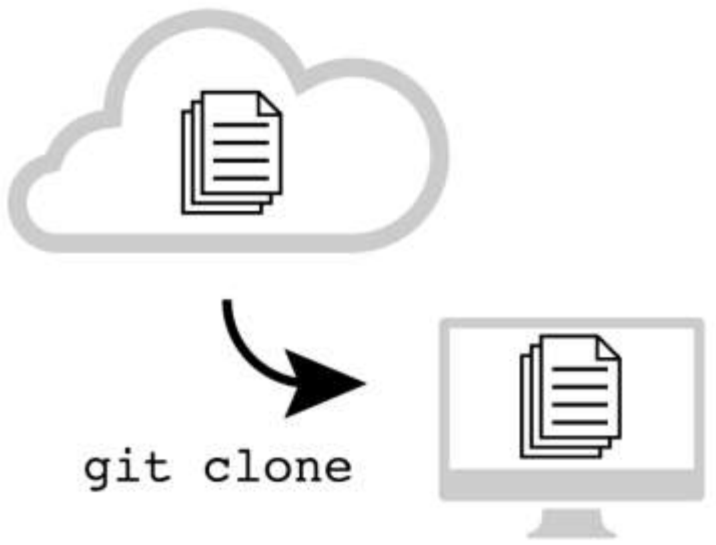
Location: C:\Users\RileyW\.gitconfig [Open]

Key	Value
apply	
branch	
autosetuprebase	true
core	
autocrlf	true
gui	
recentrepo	C:/git/smartManualTemp
http	

Cloning a Repository

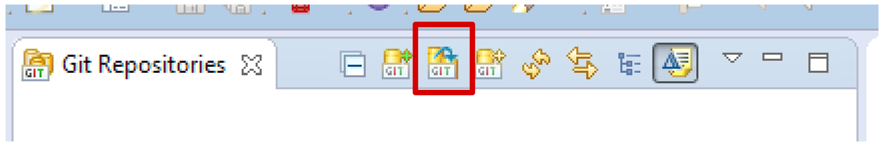
Cloning a repository

- Cloning is the Git equivalent to a checkout in SVN
- You get a complete copy of the repository including the full history
- By default a branch is checked out into your working copy
- You cannot just clone a subsection of the repository

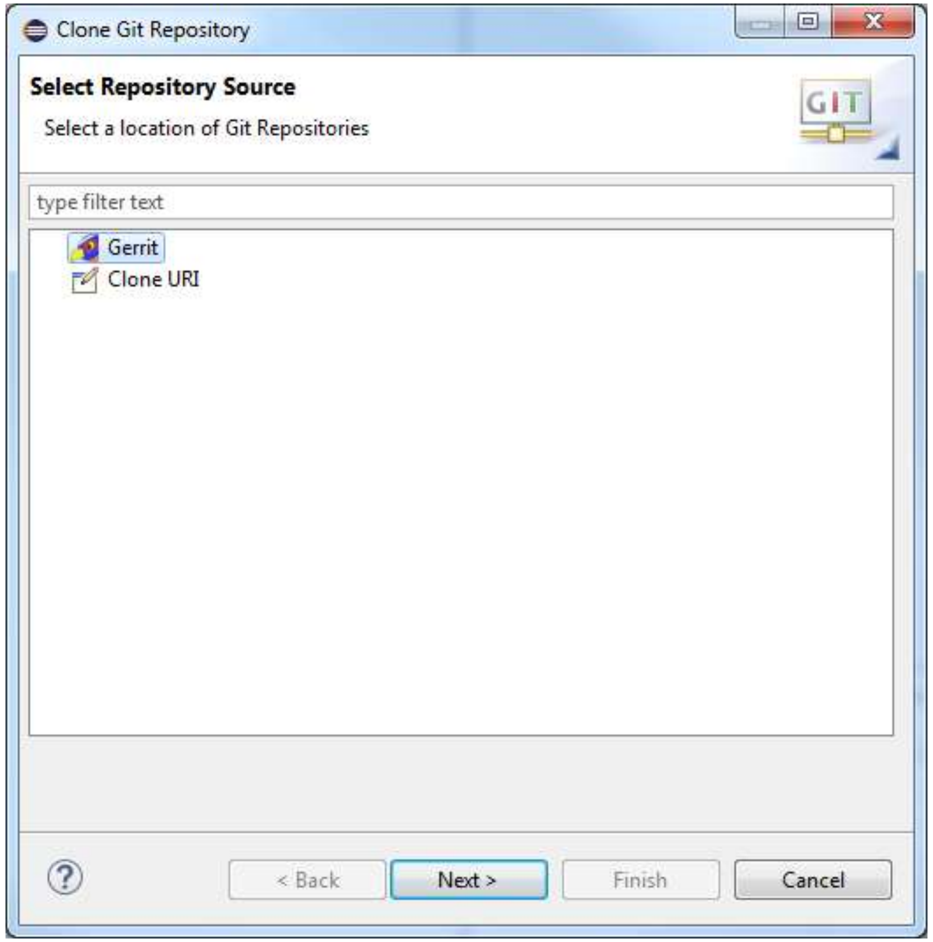


Cloning a repository (1/6)

- Go to the “Git” perspective
- In the “Git Repositories” view click “Clone”



- Select “Gerrit”

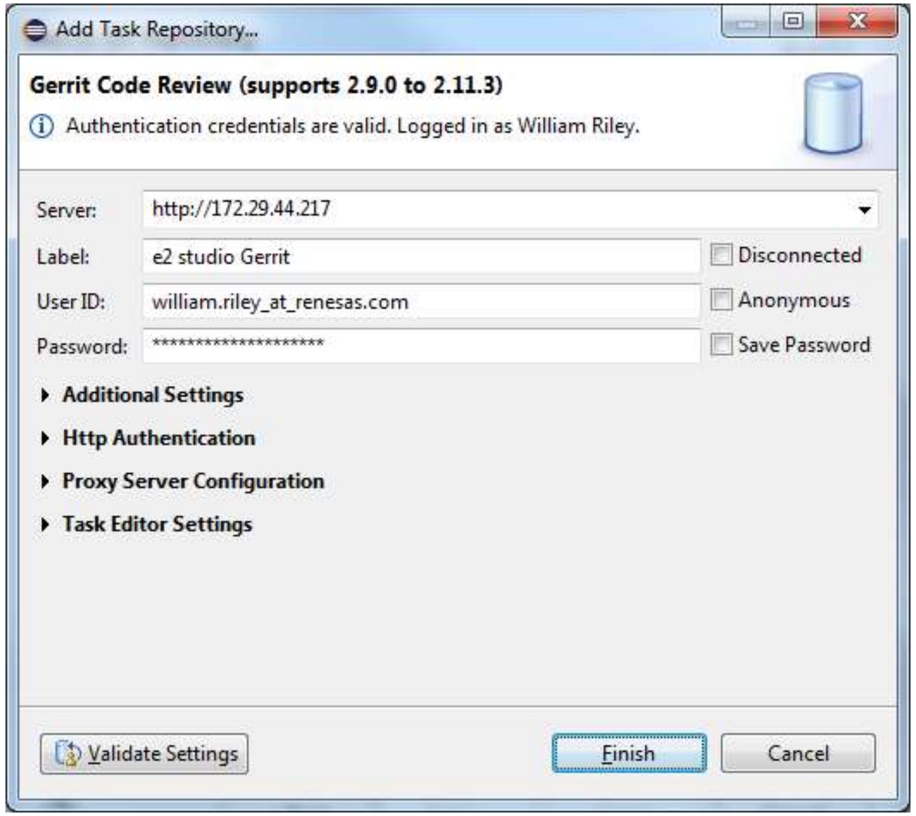


Cloning a repository (2/6)

If you have not cloned a repository before you will need to set up Eclipse to communicate with Gerrit

1. Click Add
2. Enter Settings:
   1. Server: http://172.29.44.217
   2. Label: “e2 studio Gerrit”
   3. User ID: JIRA username
   4. Password: JIRA password

- Click Validate Settings
- If successful you will see the following:

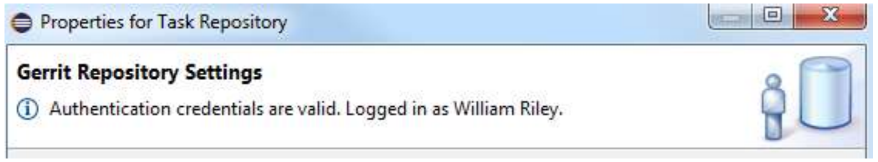


Add Task Repository...

Gerrit Code Review (supports 2.9.0 to 2.11.3)

Authentication credentials are valid. Logged in as William Riley.

Server:	http://172.29.44.217	
Label:	e2 studio Gerrit	Disconnected
User ID:	william.riley_at_renesas.com	Anonymous
Password:	********************	Save Password



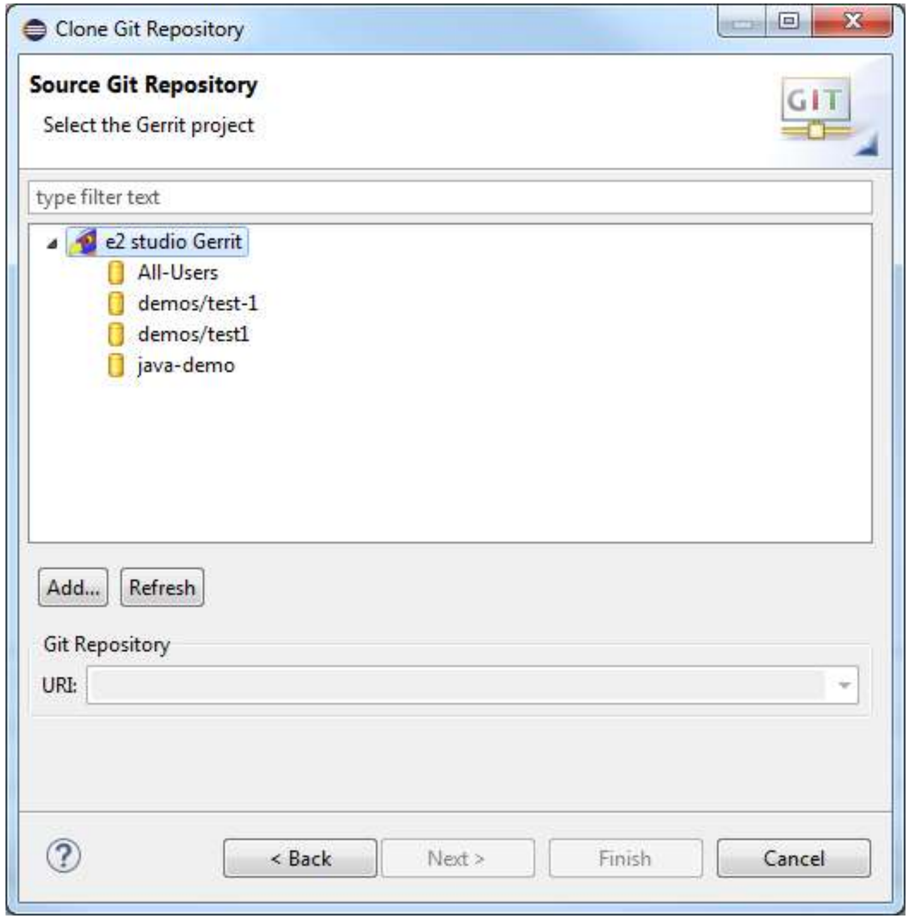
Properties for Task Repository

Gerrit Repository Settings

Authentication credentials are valid. Logged in as William Riley.

Cloning a repository (3/6)

1. Select “e2 studio Gerrit”
2. Click “Refresh”
3. Expand the list
4. Select the repository you wish to clone
   – For this training select “Demo-Projects/eclipse-plugin-demo”
5. Click Next


Clone Git Repository

Source Git Repository

Select the Gerrit project

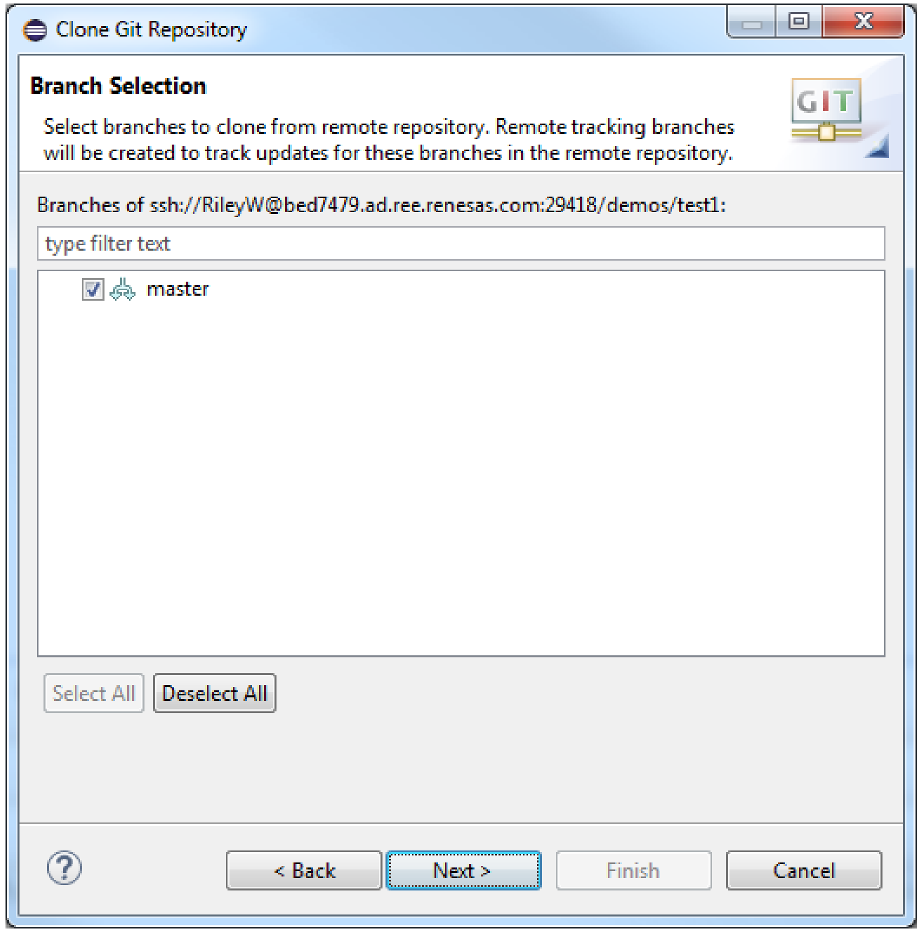
type filter text

- e2 studio Gerrit
  - All-Users
  - demos/test-1
  - demos/test1
  - java-demo

Cloning a repository (4/6)

1. Select the branches you wish to clone and click Next

If you see an error please check your SSH key settings



Clone Git Repository

Branch Selection

Select branches to clone from remote repository. Remote tracking branches will be created to track updates for these branches in the remote repository.

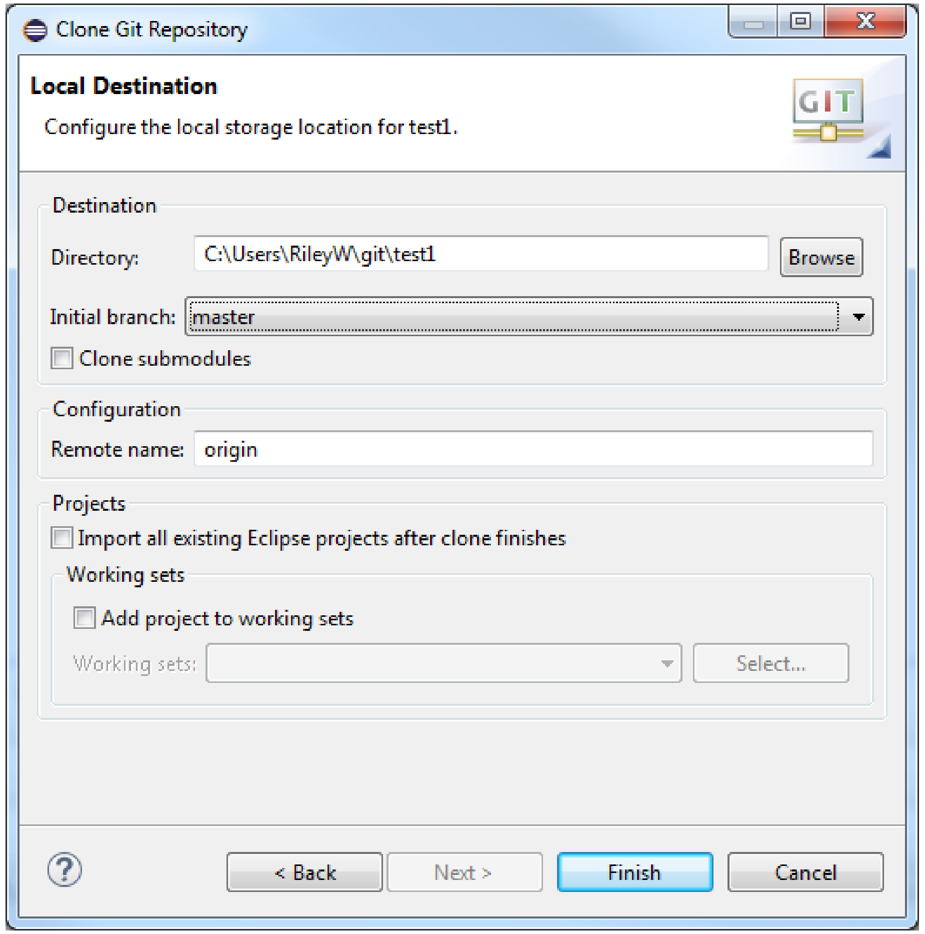
Branches of ssh://RileyW@bed7479.ad.ree.renesas.com:29418/demos/test1:

type filter text

☑ master

Cloning a repository (5/6)

1. Specify the location you wish to clone to
2. Select the initial branch, this will usually be “master”
3. Leave “Remote Name” as “origin”
4. Click “Finish”



Clone Git Repository

Local Destination

Configure the local storage location for test1.

Destination

Directory: C:\Users\RileyW\git\test1

Initial branch: master

☐ Clone submodules

Configuration

Remote name: origin

Projects

☐ Import all existing Eclipse projects after clone finishes

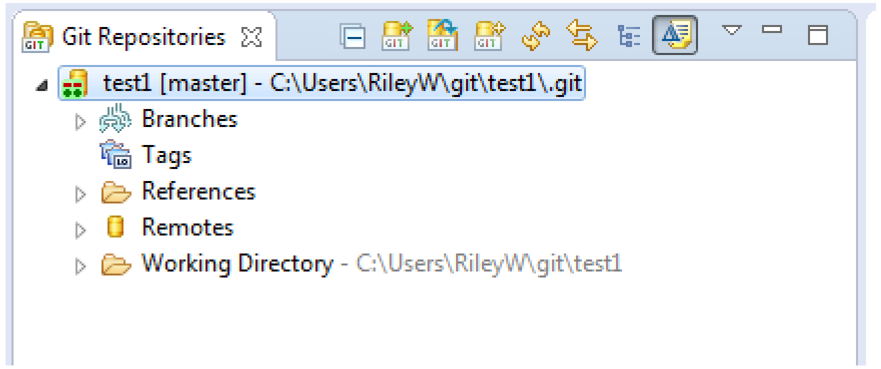
Working sets

☐ Add project to working sets

Working sets:

Cloning a repository (6/6)

- If successful you will now see the repository listed in the “Git Repositories” view



```
test1 [master] - C:\Users\RileyW\git\test1\.git
    Branches
    Tags
    References
    Remotes
    Working Directory - C:\Users\RileyW\git\test1
```

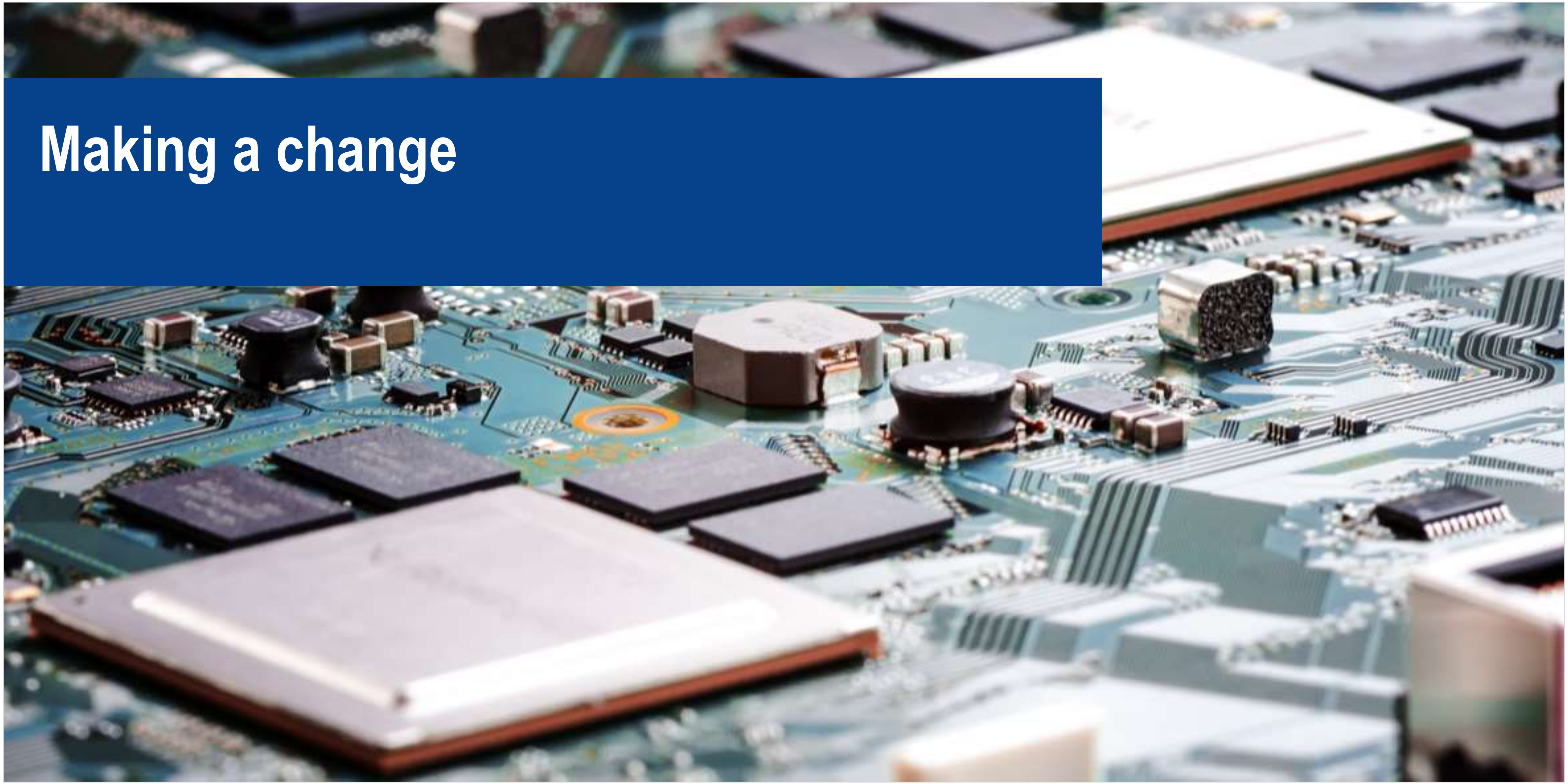
- Please check for any instructions on https://jira.renesas.eu/confluence/x/n4Ah
- If you are developing Eclipse plugins you can right click to import projects into your workspace

Getting Updates

There are 2 type of update you can do in Git

- Fetch – This just gets any changes on a remote repository but does not update any local branches.
- Pull – Same as fetch but also merges/rebases any remote branch changes into the current local branch

- Use a Fetch when preparing to create a new local branch or preparing for a rebase
- Use Pull to update your current local branch



Making a change

Making a change (1/9)

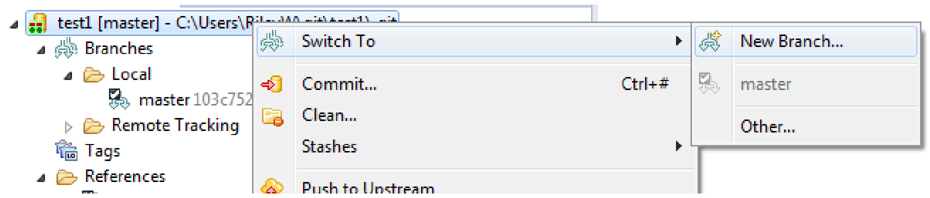
The 1st step for making a new change is to update your copy of the repository

1. Make sure you are on the correct branch. If not switch to it.
2. Right click on the repository and click “Pull”

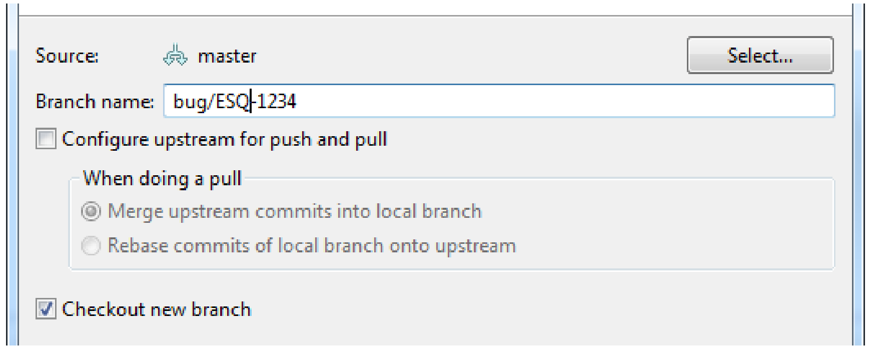
Making a change (2/9)

You must create a new branch, if you don't you will have issues updating in future

1. Right click on the repository
2. Go to "switch to -> New Branch"



3. Enter the new branch name, suggest "bug/<ticket-id>" eg. "bug/IDE-1234"
4. Click finish


Source: master [Select...]

Branch name: bug/ESQ-1234

☐ Configure upstream for push and pull

When doing a pull
◉ Merge upstream commits into local branch
○ Rebase commits of local branch onto upstream

☑ Checkout new branch

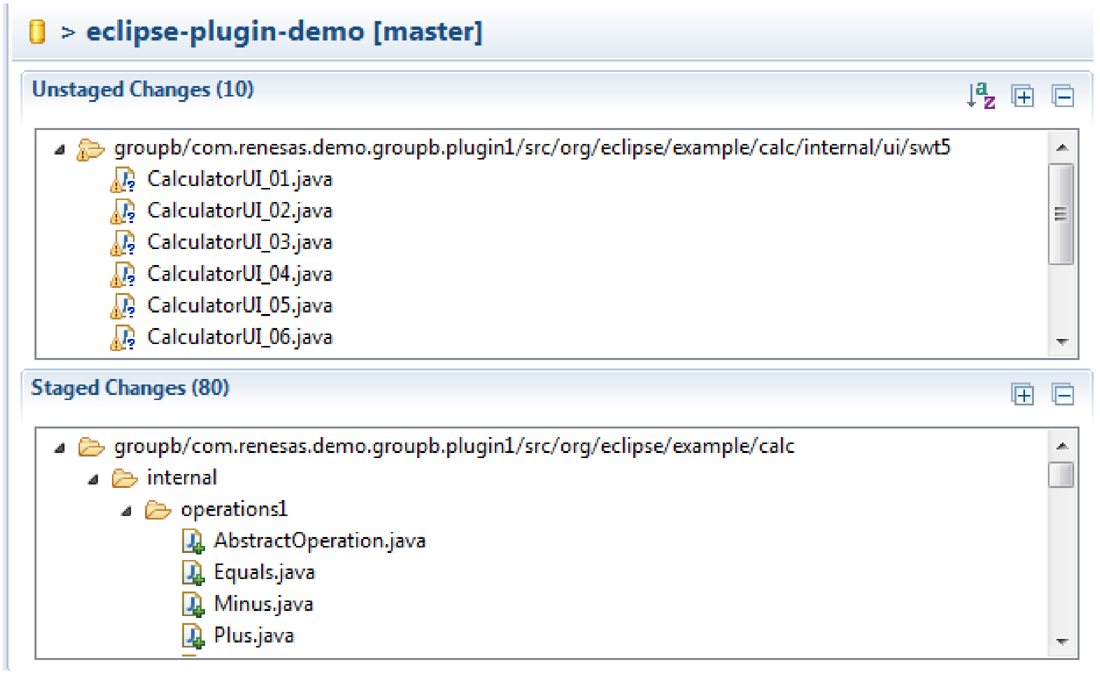
Making a change (3/9)

Make your change…

Making a change (4/9)

Now you need to commit your changes

1. Go to the Git perspective
2. Open the “Git Staging” view
3. Click the repository in “Git Repositories”
4. Select your files in “Upstaged Changes”
5. Right click and “Add to Index” or drag to “Staged Changes”
6. Only staged files will be committed



> eclipse-plugin-demo [master]

Unstaged Changes (10)

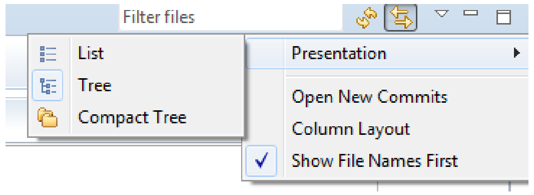
```
groupb/com.renesas.demo.groupb.plugin1/src/org/eclipse/example/calc/internal/ui/swt5
    CalculatorUI_01.java
    CalculatorUI_02.java
    CalculatorUI_03.java
    CalculatorUI_04.java
    CalculatorUI_05.java
    CalculatorUI_06.java
```

Staged Changes (80)

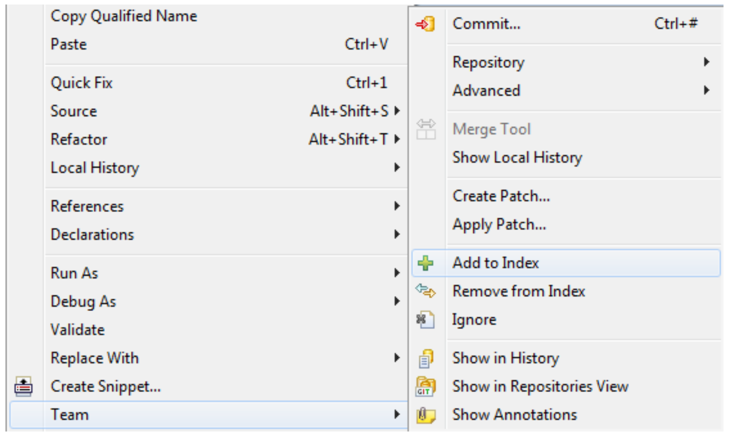
```
groupb/com.renesas.demo.groupb.plugin1/src/org/eclipse/example/calc
    internal
        operations1
            AbstractOperation.java
            Equals.java
            Minus.java
            Plus.java
```

Making a change (5/9)

- You can change the view of changed files in the view menu



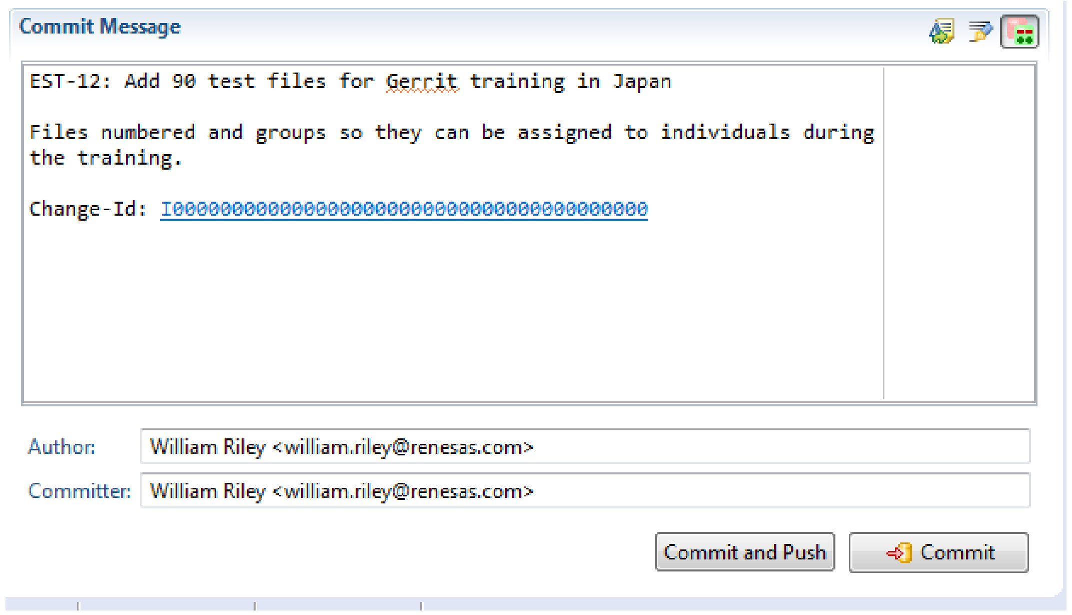
- If you are developing in Eclipse you can stage changes as you go by right clicking on the file in the Package Explorer or Editor. Once a change is staged you will need to stage again if you make more changes



Making a change (6/9)

Now you need to commit your changes

1. Write your commit message
   1. We have new rules for commit messages. Some parts are enforced, you will not be able to push your change if you do not comply.
2. Make sure “Add Change-Id” is selected
3. Check the author and committer fields
   1. These should be “Your Name <youremail>”.
   2. Gerrit will reject your change if these are incorrect
4. Click Commit
5. You change is now committed to your local repository only



```
EST-12: Add 90 test files for Gerrit training in Japan

Files numbered and groups so they can be assigned to individuals during
the training.

Change-Id: I0000000000000000000000000000000000000000
```

Author: William Riley <william.riley@renesas.com>

Committer: William Riley <william.riley@renesas.com>

Commit message rules

- See below
- Ticket ID presence is enforced

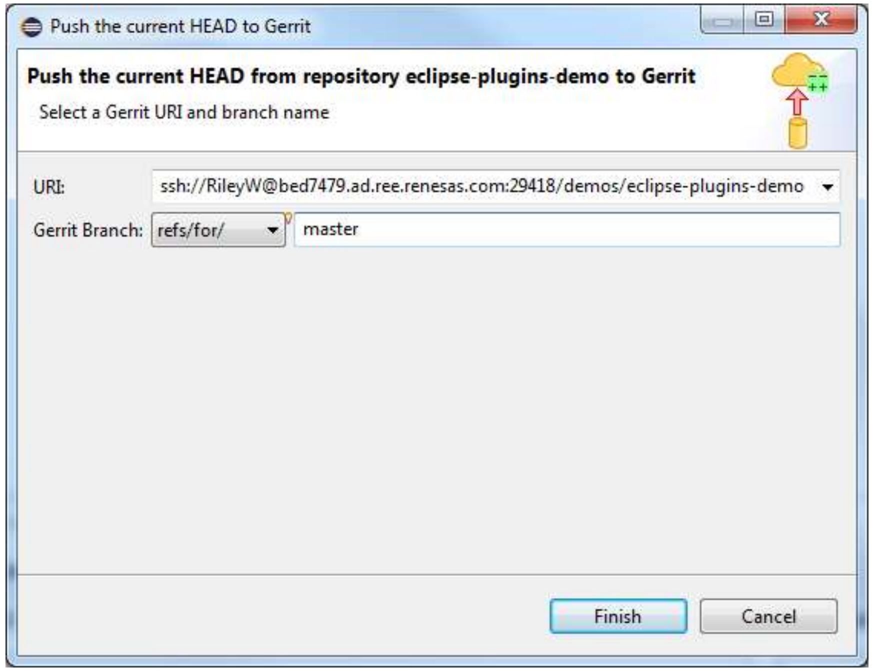
```
IDE-1234: Short (Between 5 and 50 chars) summary of changes

More detailed explanatory text, if necessary.  Wrap it to
about 72 characters or so.  In some contexts, the first
line is treated as the subject of an email and the rest of
the text as the body.  The blank line separating the
summary from the body is critical (unless you omit the body
entirely); tools like rebase can get confused if you run
the two together.

Further paragraphs come after blank lines.

- Bullet points are okay, too

- Use a hyphen for the bullet, preceded by a single space,
with blank lines in between.
```

```
IDE-7584: Use Mars in target platform

Update target platform to use Eclipse Mars (4.5.0) release.
Also update version numbers to 5.0.0.qualifier
```

Making a change (7/9)

Finally you need to push your change to Gerrit

1. Right click on the repository in the repository view
2. Click “Push to Gerrit…”
3. Check “refs/for/” is selected
4. Enter the name of the branch you are pushing to, usually “master”
5. Click Finish



Push the current HEAD from repository eclipse-plugins-demo to Gerrit

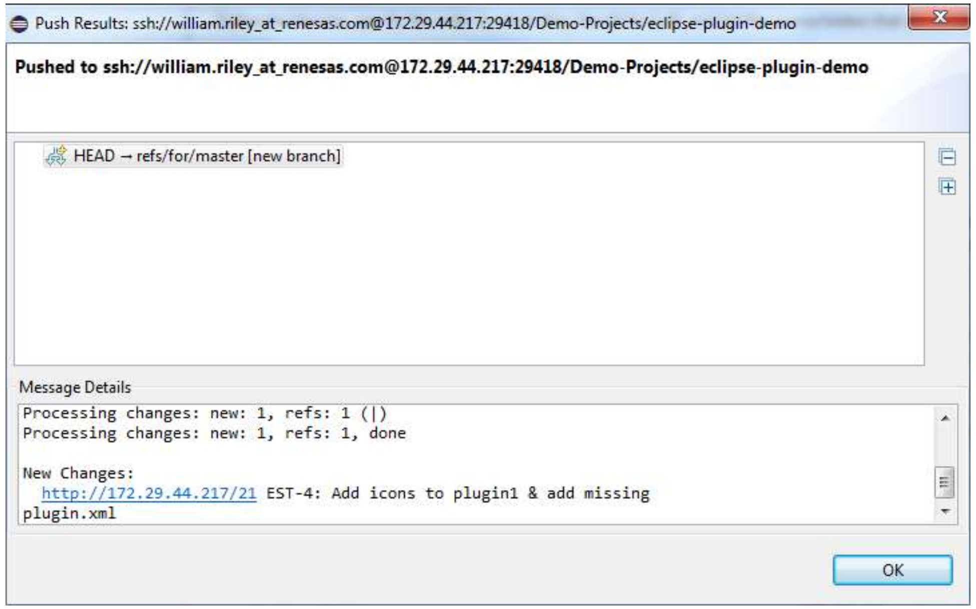
Select a Gerrit URI and branch name

URI: ssh://RileyW@bed7479.ad.ree.renesas.com:29418/demos/eclipse-plugins-demo

Gerrit Branch: refs/for/ master

Making a change (8/9)

- You will now see a results dialog.
- If everything worked you will see similar to the below.
- If there was an error check the “Message Details” for more information

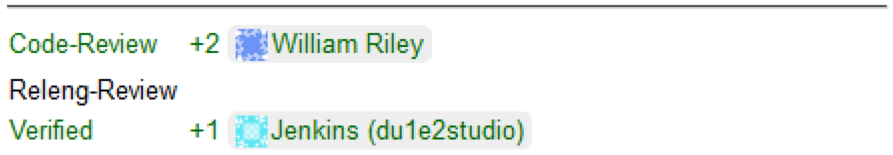

Push Results: ssh://william.riley_at_renesas.com@172.29.44.217:29418/Demo-Projects/eclipse-plugin-demo
HEAD → refs/for/master [new branch]
Message Details
OK

Pushed to ssh://william.riley_at_renesas.com@172.29.44.217:29418/Demo-Projects/eclipse-plugin-demo

```
Processing changes: new: 1, refs: 1 (|)
Processing changes: new: 1, refs: 1, done

New Changes:
  http://172.29.44.217/21 EST-4: Add icons to plugin1 & add missing
plugin.xml
```

Making a change (9/9)

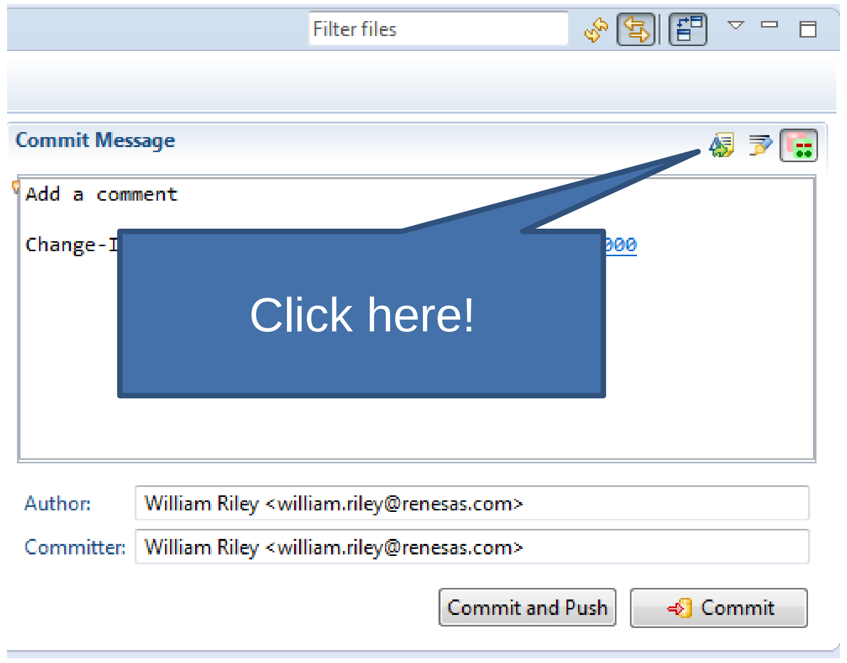
- Now you should add reviewers to your change
- Gerrit will add some automatically based on the committed file location. Check if you need to add more.
- If you commit files requiring Releng or Repo Admin reviews you may need to add the appropriate reviewers. See the list on: https://jira.renesas.eu/confluence/x/Y4Ah



Code-Review	+2	William Riley
Releng-Review		
Verified	+1	Jenkins (du1e2studio)

What if you have to amend your commit?

- On the Staging view select “Amend”
- Will automatically populate the commit message
- Stage your new changes
- Click commit
- Push again



```
Add a comment

Change-I                                              000
```

Author: William Riley <william.riley@renesas.com>
Committer: William Riley <william.riley@renesas.com>

Reviewing a change

Reviewing a change

- The most important part of the process
- You will receive an e-mail notification from Gerrit when you are added as a reviewer.
  - You may be added automatically based on the changed files
  - Committers & reviewers can add additional reviewers
- Remember code will not be submitted into the repository until the review is completed
  - Won’t be in nightly or weekly builds
- A change may need to undergo several review rounds over multiple patchsets
  - If all changes are approved with no changes something is probably not being done correctly.

Who should review

- At least one person other than the author needs to review each change
- For large changes multiple reviewers are recommended
- If a change crosses into code which is another teams responsibility someone from that team should review as well (*except for build file only changes)
- Gerrit will assign reviewers automatically based on changed files, these will be the “+2” reviewers for the team responsible for that part of the code
  - For reviews from outside their team they may delegate to another team member
  - For internal changes each team should have rules for assigning reviewers for internal changes
  - Full details of suggested practice will be found:

What should be reviewed

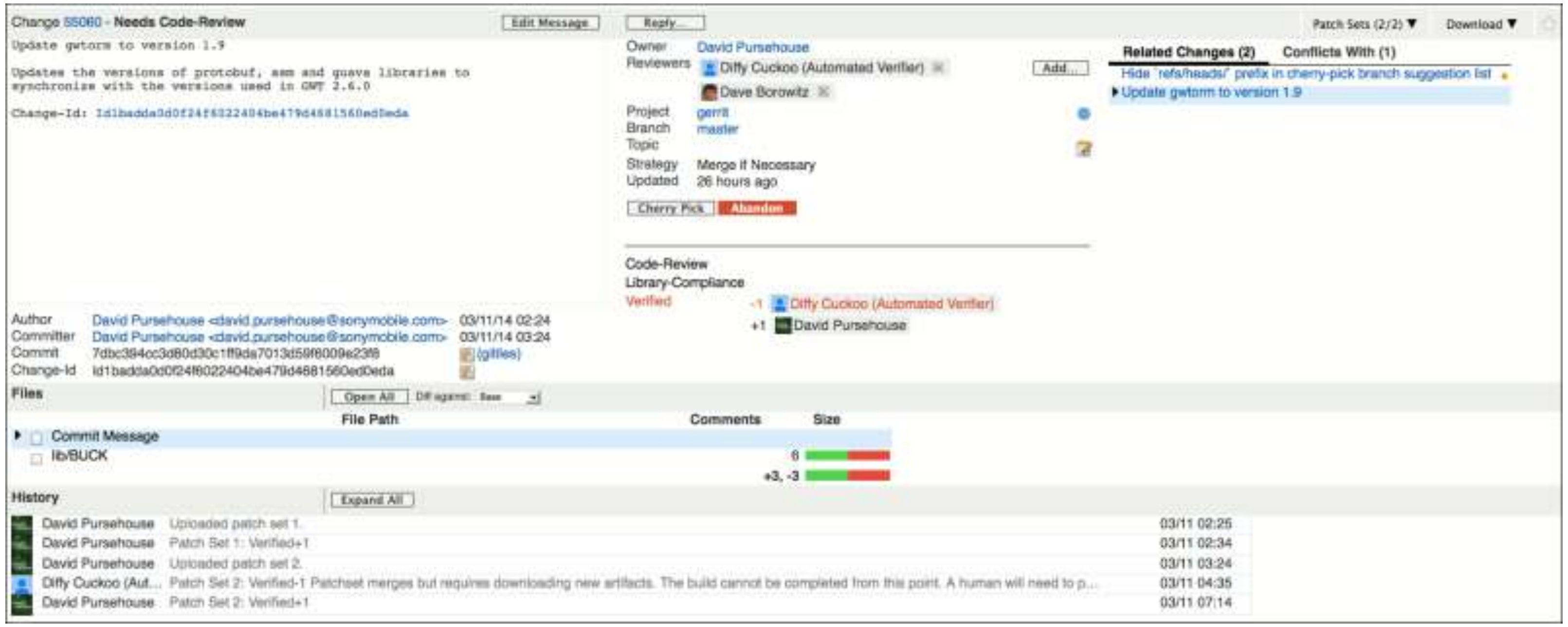
- The Gerrit review should replace the code review which would normally be done post development.
- Needs to cover all aspects of the change
- Basic checklist:
  - Should check meets coding standards
  - Doesn’t change unrelated code
  - Actually fixes the issue (pulling out change locally if required)
- More complete checklist can be found on the e2 studio Confluence Space
- Teams are welcome to add extra items for their own reviews

Gerrit Change UI

- This will give you a basic overview
- For more detail see: http://172.29.44.217/Documentation/user-review-ui.html

Change Screen

- The Gerrit change screen shows the details of a change as is the starting point for the review process:



Change 55060 - **Needs Code-Review**

```
Update gwtorm to version 1.9

Updates the versions of protobuf, asm and guava libraries to
synchronize with the versions used in GWT 2.6.0

Change-Id: Id1badda0d0f24f6022404be479d4881560ed0eda
```

Owner	David Pursehouse
Reviewers	Diffy Cuckoo (Automated Verifier)
	Dave Borowitz
Project	gerrit
Branch	master
Topic	
Strategy	Merge if Necessary
Updated	26 hours ago

Related Changes (2) Conflicts With (1)

Hide `refs/heads/` prefix in cherry-pick branch suggestion list

Update gwtorm to version 1.9

Code-Review		
Library-Compliance		
Verified	-1	Diffy Cuckoo (Automated Verifier)
	+1	David Pursehouse

Author	David Pursehouse <david.pursehouse@sonymobile.com>	03/11/14 02:24
Committer	David Pursehouse <david.pursehouse@sonymobile.com>	03/11/14 03:24
Commit	7dbc394cc3d80d30c1ff9da7013d59f6009e23f6	(gitiles)
Change-Id	Id1badda0d0f24f6022404be479d4881560ed0eda	

Files

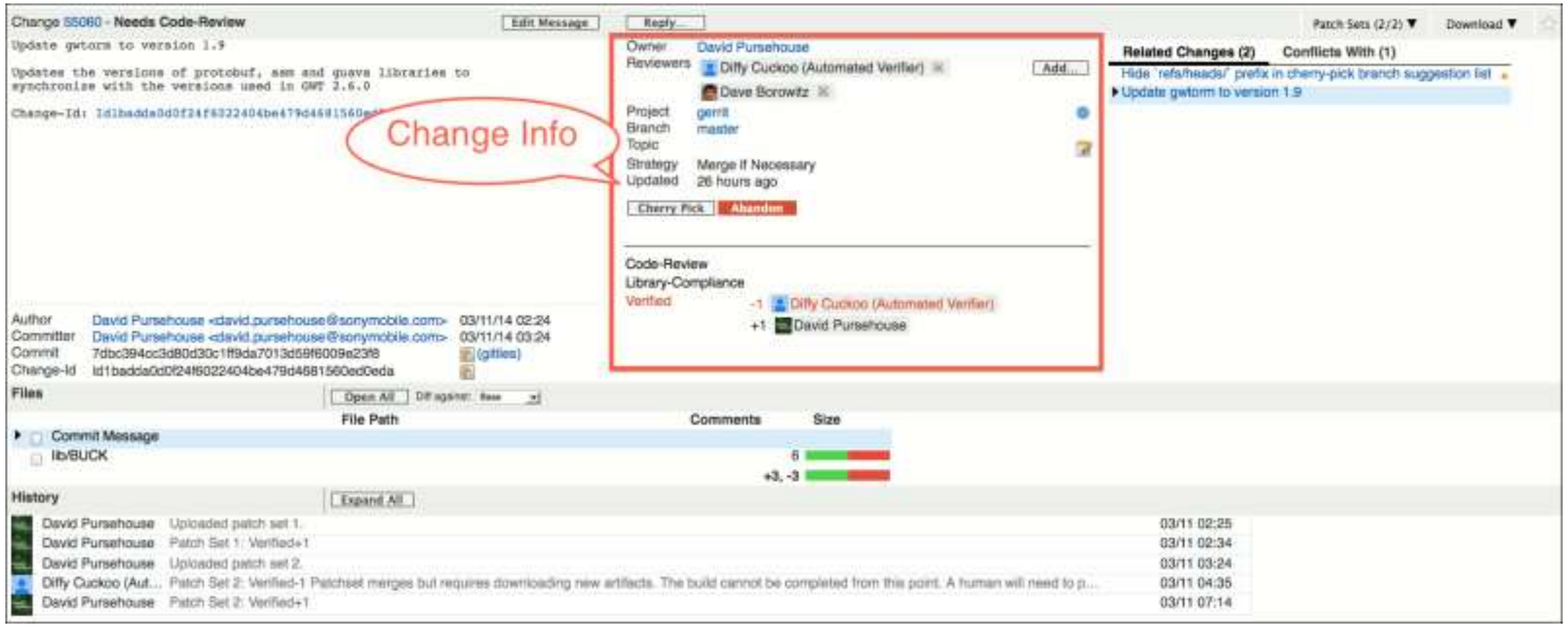
File Path	Comments	Size
Commit Message		6
lib/BUCK		+3, -3

History

David Pursehouse	Uploaded patch set 1.	03/11 02:25
David Pursehouse	Patch Set 1: Verified+1	03/11 02:34
David Pursehouse	Uploaded patch set 2.	03/11 03:24
Diffy Cuckoo (Aut...	Patch Set 2: Verified-1 Patchset merges but requires downloading new artifacts. The build cannot be completed from this point. A human will need to p...	03/11 04:35
David Pursehouse	Patch Set 2: Verified+1	03/11 07:14

Change Info Block

- This block contains detailed information on the current status of the change
  This is where you add reviewers,



Change 55060 - Needs Code-Review

```
Update gwtorm to version 1.9

Updates the versions of protobuf, asm and guava libraries to
synchronize with the versions used in GWT 2.6.0

Change-Id: Id1badda0d0f24f6022404be479d4681560e
```

Owner	David Pursehouse
Reviewers	Diffy Cuckoo (Automated Verifier)
	Dave Borowitz
Project	gerrit
Branch	master
Topic	
Strategy	Merge If Necessary
Updated	26 hours ago

Code-Review
Library-Compliance
Verified -1 Diffy Cuckoo (Automated Verifier)
+1 David Pursehouse

Related Changes (2) Conflicts With (1)
Hide `refs/heads/` prefix in cherry-pick branch suggestion list
Update gwtorm to version 1.9

Author	David Pursehouse <david.pursehouse@sonymobile.com>	03/11/14 02:24
Committer	David Pursehouse <david.pursehouse@sonymobile.com>	03/11/14 03:24
Commit	7dbc394cc3d80d30c1ff9da7013d59f6009e23f8	(gitiles)
Change-Id	Id1badda0d0f24f6022404be479d4681560ed0eda	

Files

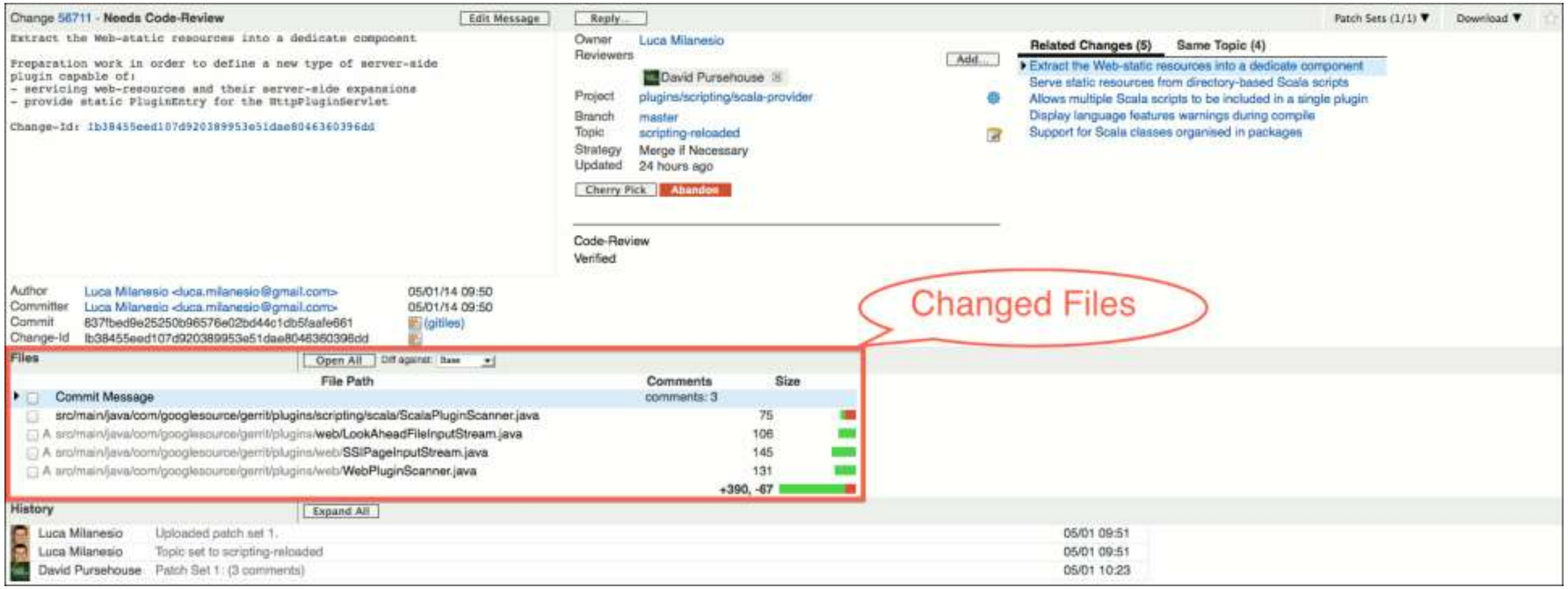
File Path	Comments	Size
Commit Message		
lib/BUCK		6
		+3, -3

History

David Pursehouse	Uploaded patch set 1.	03/11 02:25
David Pursehouse	Patch Set 1: Verified+1	03/11 02:34
David Pursehouse	Uploaded patch set 2.	03/11 03:24
Diffy Cuckoo (Aut...	Patch Set 2: Verified-1 Patchset merges but requires downloading new artifacts. The build cannot be completed from this point. A human will need to p...	03/11 04:35
David Pursehouse	Patch Set 2: Verified+1	03/11 07:14

File List

- This block contains information on all the files effected by the change.



Change 56711 - **Needs Code-Review**

```
Extract the Web-static resources into a dedicate component

Preparation work in order to define a new type of server-side
plugin capable of:
- servicing web-resources and their server-side expansions
- provide static PluginEntry for the HttpPluginServlet

Change-Id: 1b38455eed107d920389953e51dae8046360396dd
```

Owner	Luca Milanesio
Reviewers	David Pursehouse
Project	plugins/scripting/scala-provider
Branch	master
Topic	scripting-reloaded
Strategy	Merge if Necessary
Updated	24 hours ago

Code-Review
Verified

Related Changes (5) Same Topic (4)

- Extract the Web-static resources into a dedicate component
- Serve static resources from directory-based Scala scripts
- Allows multiple Scala scripts to be included in a single plugin
- Display language features warnings during compile
- Support for Scala classes organised in packages

Author	Luca Milanesio <luca.milanesio@gmail.com>	05/01/14 09:50
Committer	Luca Milanesio <luca.milanesio@gmail.com>	05/01/14 09:50
Commit	837fbed9e25250b96576e02bd44c1db5faafe661	(gitiles)
Change-Id	Ib38455eed107d920389953e51dae8046360396dd	

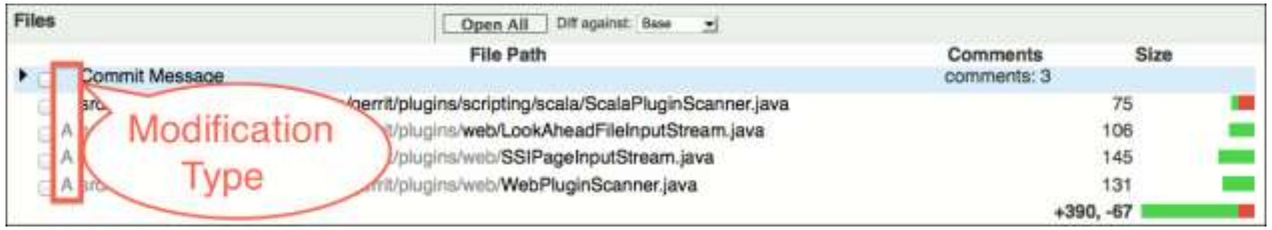
Files

File Path	Comments	Size
Commit Message	comments: 3	
src/main/java/com/googlesource/gerrit/plugins/scripting/scala/ScalaPluginScanner.java		75
A src/main/java/com/googlesource/gerrit/plugins/web/LookAheadFileInputStream.java		106
A src/main/java/com/googlesource/gerrit/plugins/web/SSIPageInputStream.java		145
A src/main/java/com/googlesource/gerrit/plugins/web/WebPluginScanner.java		131
		+390, -67

History

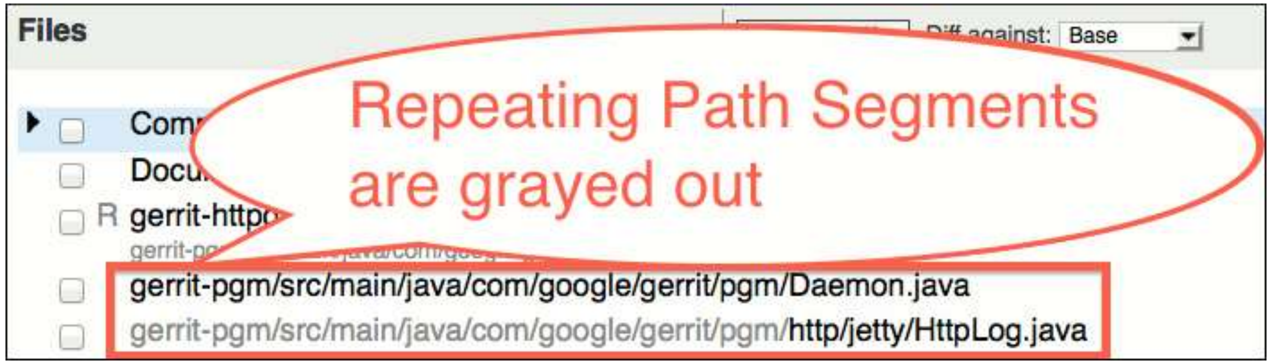
Luca Milanesio	Uploaded patch set 1.	05/01 09:51
Luca Milanesio	Topic set to scripting-reloaded	05/01 09:51
David Pursehouse	Patch Set 1: (3 comments)	05/01 10:23

File List



Files — Open All — Diff against: Base

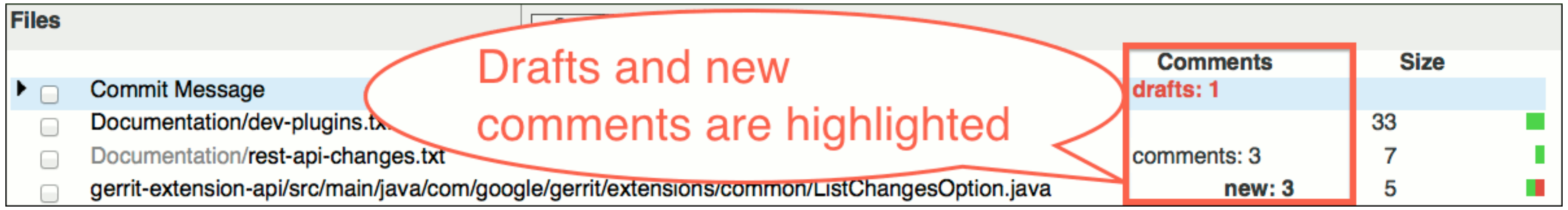
	File Path	Comments	Size
	Commit Message	comments: 3	
	src… /gerrit/plugins/scripting/scala/ScalaPluginScanner.java		75
A	…t/plugins/web/LookAheadFileInputStream.java		106
A	…/plugins/web/SSIPageInputStream.java		145
A	src… rrit/plugins/web/WebPluginScanner.java		131
			+390, -67



Files — Diff against: Base

- Com…
- Docu…
- R gerrit-httpd…
  gerrit-p… java/com/goo…
- gerrit-pgm/src/main/java/com/google/gerrit/pgm/Daemon.java
- gerrit-pgm/src/main/java/com/google/gerrit/pgm/http/jetty/HttpLog.java

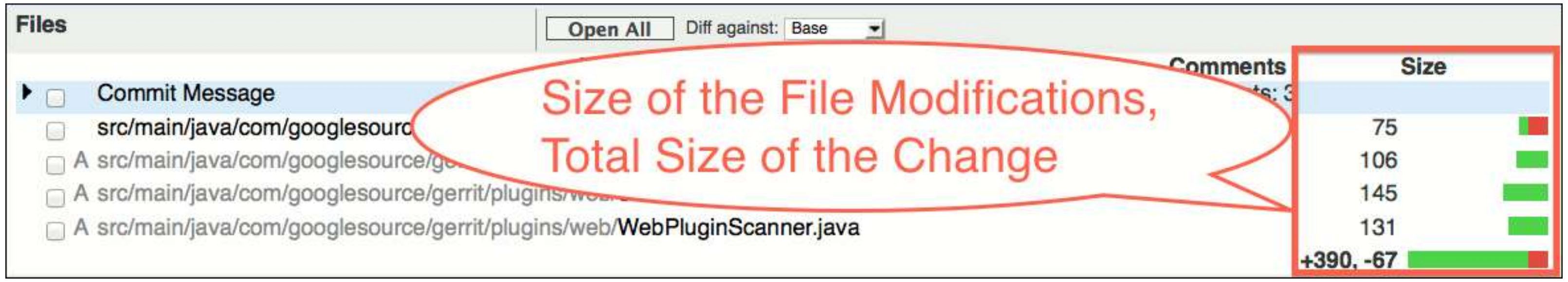
File List



Files

Drafts and new comments are highlighted

		Comments	Size
▸ ☐	Commit Message	drafts: 1	
☐	Documentation/dev-plugins.tx		33
☐	Documentation/rest-api-changes.txt	comments: 3	7
☐	gerrit-extension-api/src/main/java/com/google/gerrit/extensions/common/ListChangesOption.java	new: 3	5



Files

Open All Diff against: Base

Size of the File Modifications, Total Size of the Change

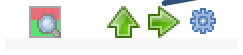
		Comments	Size
▸ ☐	Commit Message	ts: 3	
☐	src/main/java/com/googlesourc		75
☐ A	src/main/java/com/googlesource/ge		106
☐ A	src/main/java/com/googlesource/gerrit/plugins/we		145
☐ A	src/main/java/com/googlesource/gerrit/plugins/web/WebPluginScanner.java		131
			+390, -67

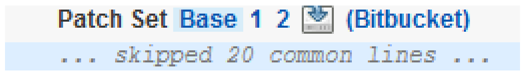
Review Process

- Start with the Commit message
- Cover all files
- Navigate between files using the Arrow icons
- Where multiple patchsets are present you can choose one to compare against
  - This allows you to easily see what changed between pathsets where the committer has made changes based on previous review comments

Click to go to the change screen

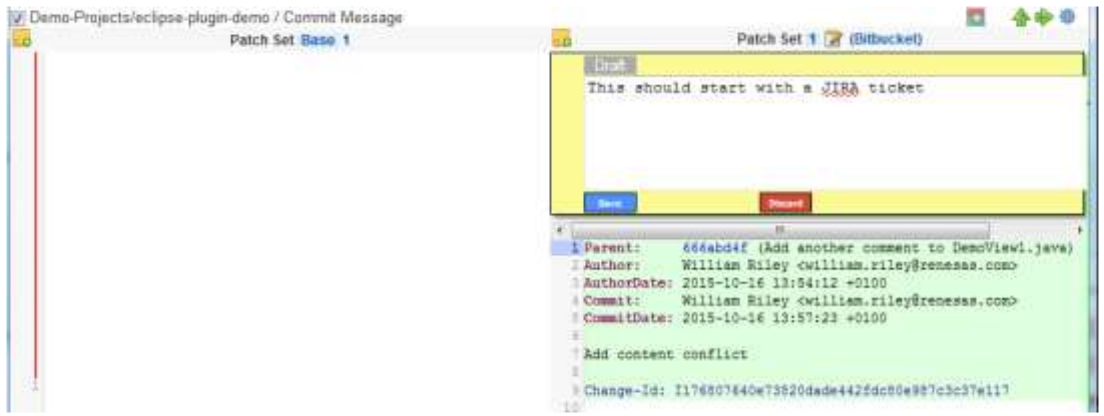
Click to go the next file





Patch Set Base 1 2 (Bitbucket)

... skipped 20 common lines ...



Demo-Projects/eclipse-plugin-demo / Commit Message

Patch Set Base 1

Patch Set 1 (Bitbucket)

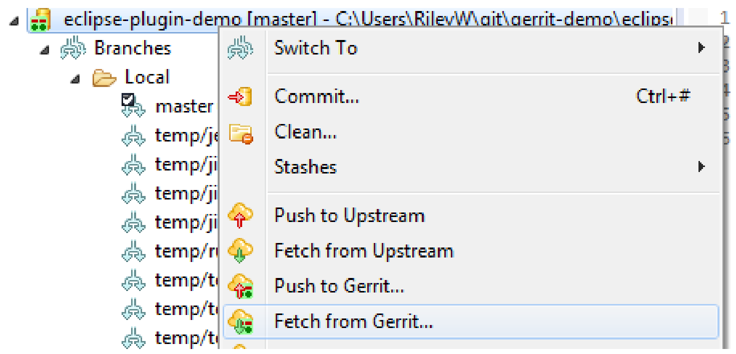
Draft

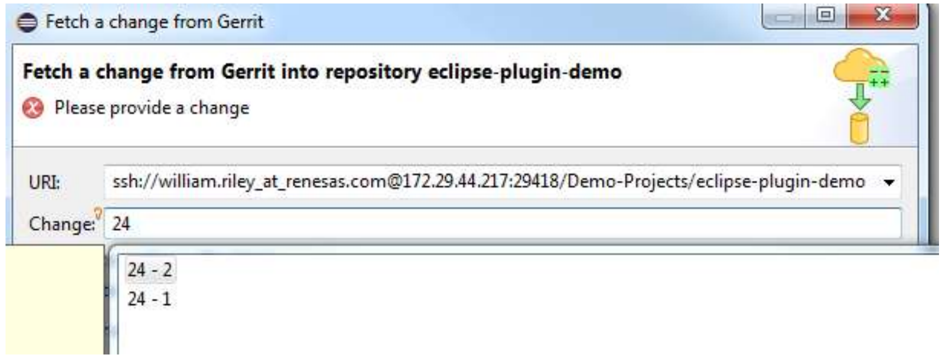
This should start with a JIRA ticket

```
Parent:     666abd4f (Add another comment to DemoView1.java)
Author:     William Riley <william.riley@renesas.com>
AuthorDate: 2015-10-16 13:54:12 +0100
Commit:     William Riley <william.riley@renesas.com>
CommitDate: 2015-10-16 13:57:23 +0100

Add content conflict

Change-Id: I176807840e73620dade442fdc80e987c3c37e117
```

Checkout the change locally

- As part of the review you may wish to test the change locally
- You can do this easily with Egit
- Select “Fetch from Gerrit”
- Enter the change number and press Ctrl + Space
- Select the patchset from the list
- Click finish




Fetch a change from Gerrit

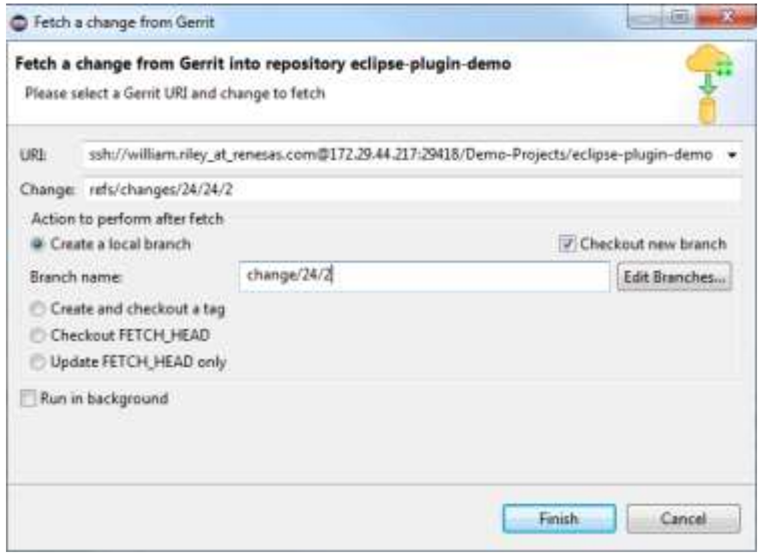
Fetch a change from Gerrit into repository eclipse-plugin-demo

Please provide a change

URI: ssh://william.riley_at_renesas.com@172.29.44.217:29418/Demo-Projects/eclipse-plugin-demo

Change: 24

24 - 2
24 - 1


Fetch a change from Gerrit

Fetch a change from Gerrit into repository eclipse-plugin-demo

Please select a Gerrit URI and change to fetch

URI: ssh://william.riley_at_renesas.com@172.29.44.217:29418/Demo-Projects/eclipse-plugin-demo

Change: refs/changes/24/24/2

Action to perform after fetch

(●) Create a local branch [✓] Checkout new branch

Branch name: change/24/2 Edit Branches...

( ) Create and checkout a tag

( ) Checkout FETCH_HEAD

( ) Update FETCH_HEAD only

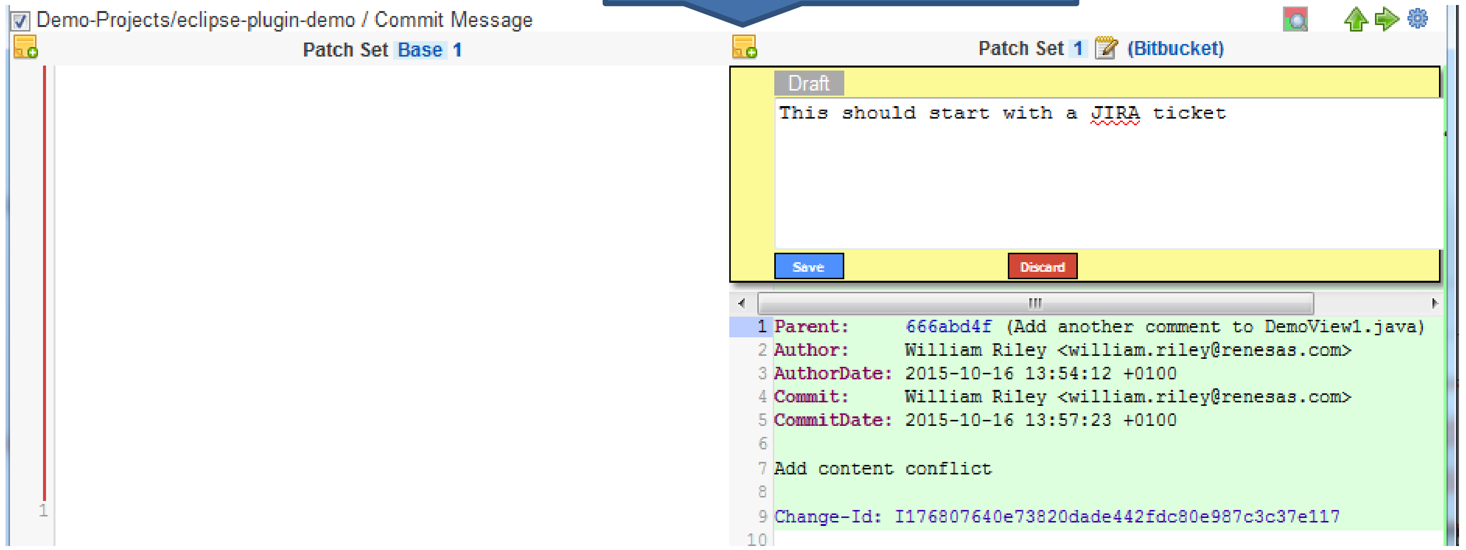
[ ] Run in background

Finish Cancel

Adding Comments

- You can add comment to a whole file by clicking the icon at the top right

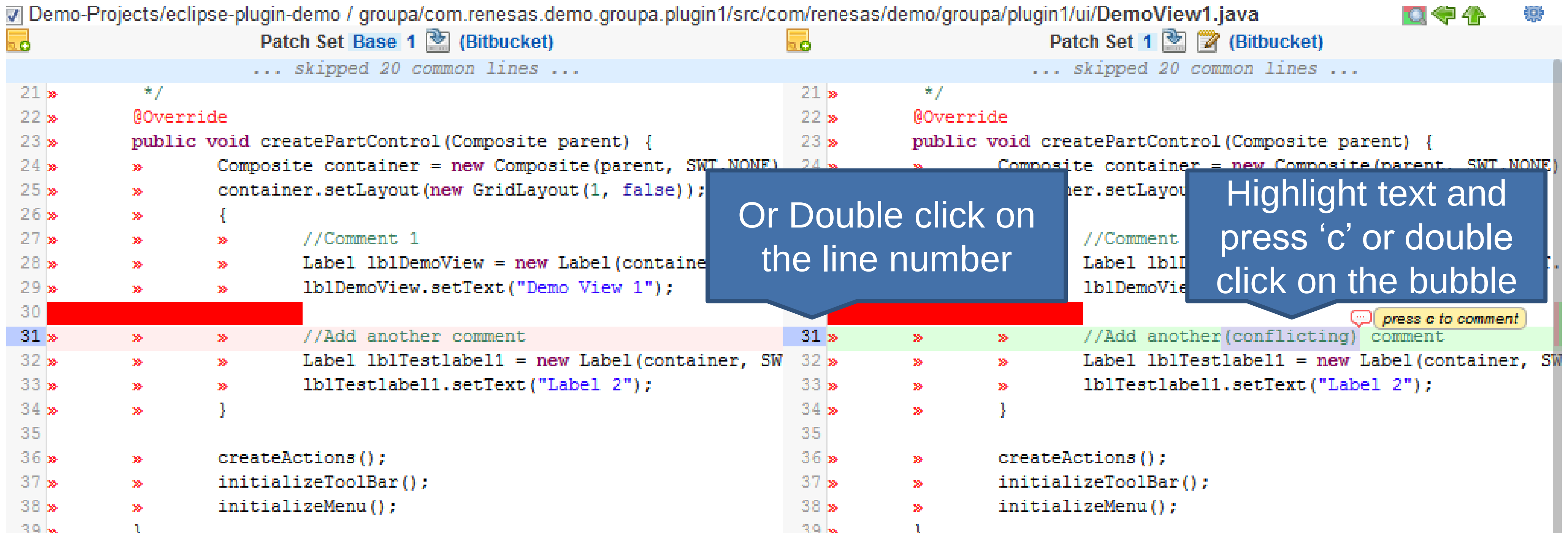
Click to add file comment



This should start with a JIRA ticket

```
Parent:     666abd4f (Add another comment to DemoView1.java)
Author:     William Riley <william.riley@renesas.com>
AuthorDate: 2015-10-16 13:54:12 +0100
Commit:     William Riley <william.riley@renesas.com>
CommitDate: 2015-10-16 13:57:23 +0100

Add content conflict

Change-Id: I176807640e73820dade442fdc80e987c3c37e117
```

Adding comments



Demo-Projects/eclipse-plugin-demo / groupa/com.renesas.demo.groupa.plugin1/src/com/renesas/demo/groupa/plugin1/ui/**DemoView1.java**

Patch Set Base 1 (Bitbucket)

```
... skipped 20 common lines ...
         */
        @Override
        public void createPartControl(Composite parent) {
                Composite container = new Composite(parent, SWT.NONE)
                container.setLayout(new GridLayout(1, false));
                {
                        //Comment 1
                        Label lblDemoView = new Label(containe
                        lblDemoView.setText("Demo View 1");

                        //Add another comment
                        Label lblTestlabel1 = new Label(container, SW
                        lblTestlabel1.setText("Label 2");
                }

                createActions();
                initializeToolBar();
                initializeMenu();
```

Patch Set 1 (Bitbucket)

```
... skipped 20 common lines ...
         */
        @Override
        public void createPartControl(Composite parent) {
                Composite container = new Composite(parent, SWT.NONE)
                        ner.setLayou

                        //Comment
                        Label lblI
                        lblDemoVie

                        //Add another (conflicting) comment
                        Label lblTestlabel1 = new Label(container, SW
                        lblTestlabel1.setText("Label 2");
                }

                createActions();
                initializeToolBar();
                initializeMenu();
```

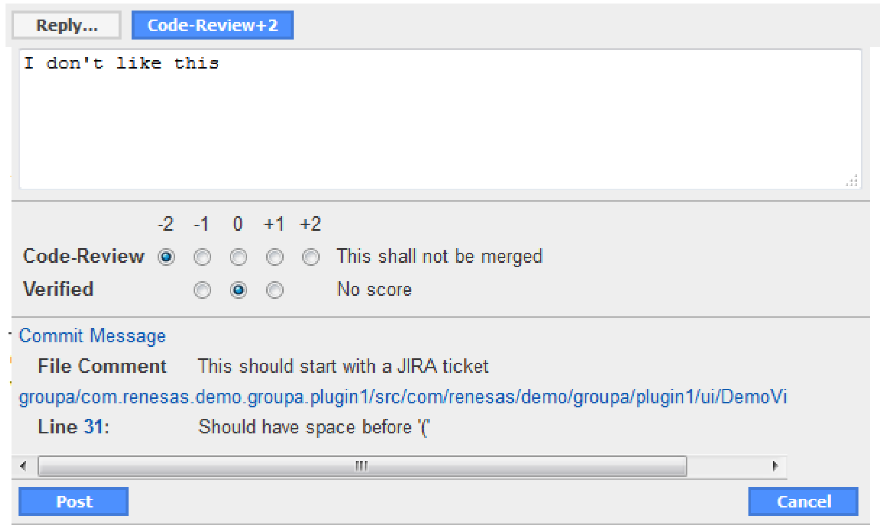
Or Double click on the line number

Highlight text and press ‘c’ or double click on the bubble

Publishing your review

Finally you need to publish your review comments

- At the top of the change screen click “Reply…”
- You should enter a summary comment for your review.
- Then select your score for “Code-Review”
  - A change requires at least one +2 code review score to be approved.
  - -2 blocks a change from submission, no matter what other scores are present
  - You can select 0 if you just wish to comment
- When done click “Post”



```
I don't like this
```

	-2	-1	0	+1	+2	
Code-Review	●	○	○	○	○	This shall not be merged
Verified		○	●	○		No score

Commit Message
File Comment This should start with a JIRA ticket
groupa/com.renesas.demo.groupa.plugin1/src/com/renesas/demo/groupa/plugin1/ui/DemoVi
Line 31: Should have space before '('

Score	Meaning
+2	Looks good to me, approved
+1	Looks good to me, but someone else must approve
0	No score
-1	I would prefer this is not merged as is
-2	This shall not be merged

Who can +2

- For Eclipse plugins each team should nominate 1 or more (depending on team size) members who will be responsible for their teams changes
  - These team members should be familiar with the code & the review policies
  - We will be contacting the managers about this
- For GDB server only Paul, Terry, Kudo-san & Kajinuma-san will be able to +2
- They should only +2 their own teams changes
  - Where a change crosses multiple teams code the originating team is responsible for the final +2
- If the author of the change can +2 they should only do so if no other team member can

When to +2

- A Code Review score of +2 in Gerrit will approve a change for submission if no other labels are needed
  - It is important this only happens once the code review is fully completed
- Each team will have a number of nominated users who have the ability to +2 changes
- They should check the following:
  - Has the review been completed?
    - All reviewers added scores, including from other teams if required
    - Comments addressed
  - Have team review policies been followed?

Additional Review Categories

- In addition to Code-Review there are a number of additional categories defined which may need to be scored before a change can be submitted
- “REE-Code-Review” – Required when a change modifies any core code. E.g. Core Eclipse plugins or Shared build plugins.
  - Can only be given by REE
- “Releng-Review” - Required for Eclipse plugins when key build files (e.g. pom.xml, manifest.mf) are modified
  - Only be given by William Riley
- “Repo-Owner-Review” – Required when key git configuration files (e.g. .gitignore) or changed or when a change modified more than 25 files.
  - William Riley or Paul Bell

Responding to a review

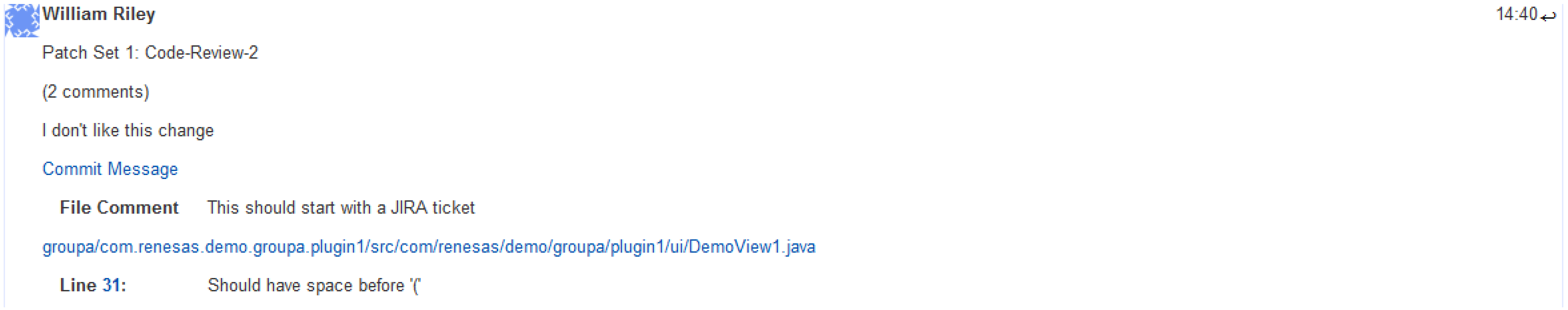
Responding to a review

- Hopefully your change will be approved and you will not need to respond
- However if a reviewer scores -1 or -2 you should check their comments
  - If -2 then your change will not proceed until they change their score

Jenkins Build

- If the Jenkins build failed due to a broken branch then In most cases you REE will retrigger any open reviews once the build is fixed.
- If this does not happen or you do not want to wait for REE to do it then you should rebase your change though Gerrit.
  - Alternatively you can rebase onto a working commit in the same branch
- If not then retrigger you should follow the link to the failed job and click Retrigger Retrigger
  - Login using your JIRA username & password

Comments from Reviewers


William Riley 14:40

Patch Set 1: Code-Review-2

(2 comments)

I don't like this change

Commit Message

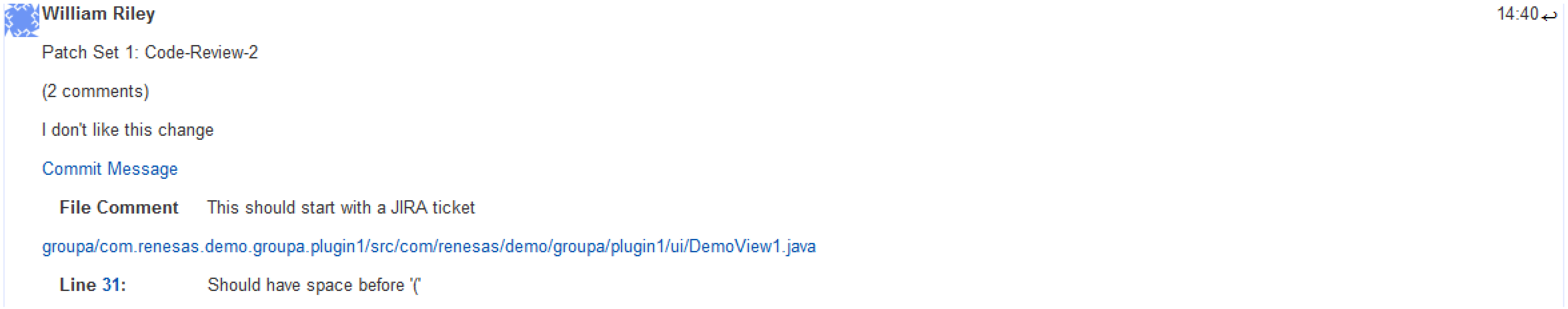
File Comment This should start with a JIRA ticket

groupa/com.renesas.demo.groupa.plugin1/src/com/renesas/demo/groupa/plugin1/ui/DemoView1.java

Line 31: Should have space before '('

Comments from Reviewers

- Click on the link in the comment to start looking at inline review comments



William Riley 14:40

Patch Set 1: Code-Review-2

(2 comments)

I don't like this change

Commit Message

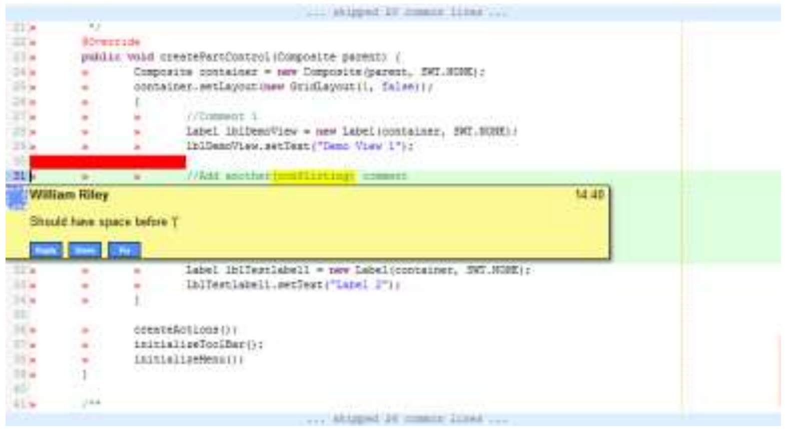
File Comment This should start with a JIRA ticket

groupa/com.renesas.demo.groupa.plugin1/src/com/renesas/demo/groupa/plugin1/ui/DemoView1.java

Line 31: Should have space before '('

Replying to inline comments

- Select the comment
  - Reply lets you post a comment replying to theirs
  - Done indicate you have accepted and fixed their comment
  - Fix allows you to fix the changing using the Gerrit editor. You may wish to do this if it is a very minor change. (Jenkins will rebuild once your new change is published).

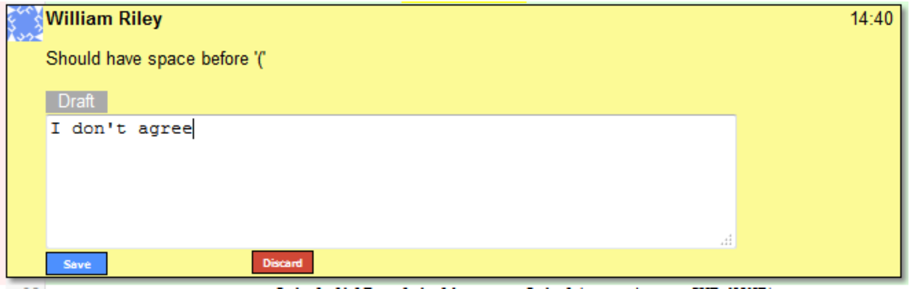


```
... skipped 20 common lines ...
    */
    @Override
    public void createPartControl(Composite parent) {
        Composite container = new Composite(parent, SWT.NONE);
        container.setLayout(new GridLayout(1, false));
        {
            //Comment 1
            Label lblDemoView = new Label(container, SWT.NONE);
            lblDemoView.setText("Demo View 1");

            //Add another(conflicting) comment
```

William Riley 14:40

Should have space before '('

```
            Label lblTestlabel1 = new Label(container, SWT.NONE);
            lblTestlabel1.setText("Label 2");
        }

        createActions();
        initializeToolBar();
        initializeMenu();
    }

    /**
... skipped 26 common lines ...
```

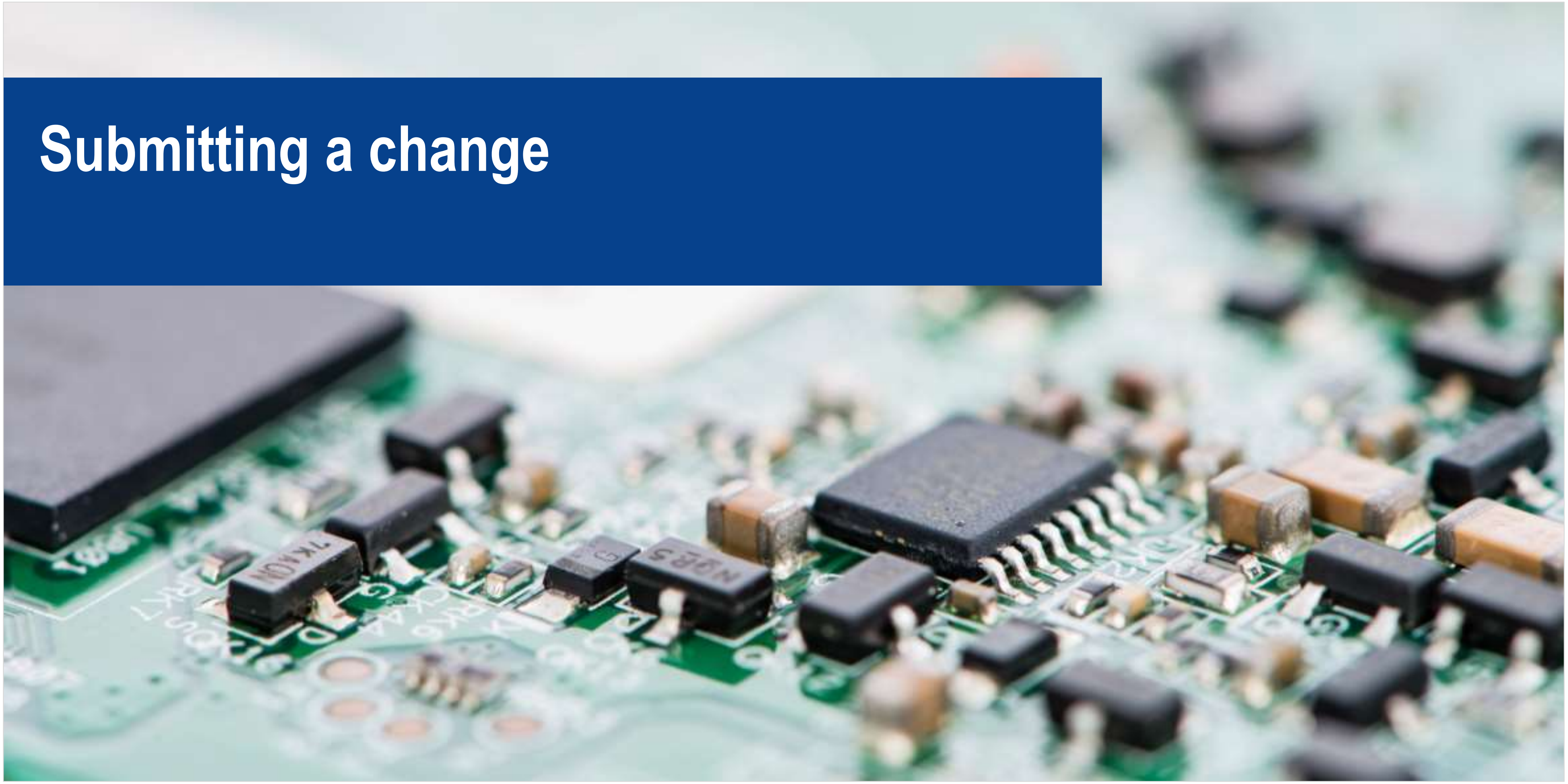


William Riley 14:40

Should have space before '('

Draft

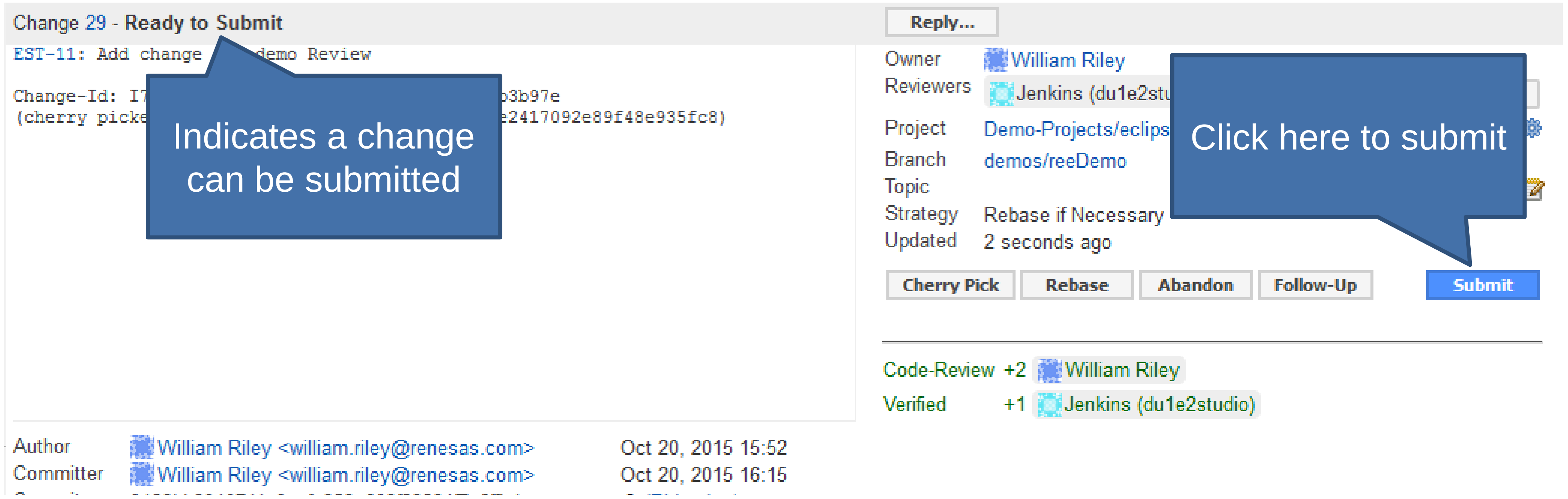
I don't agree



Submitting a change

Submitting a change

- Once a change has been approved it need to be submitted
- This is done from the change screen
- Should be done by the reviewer who gave the score which approved the change



Change 29 - Ready to Submit

```
EST-11: Add change      demo Review

Change-Id: I7                     b3b97e
(cherry picke                     e2417092e89f48e935fc8)
```

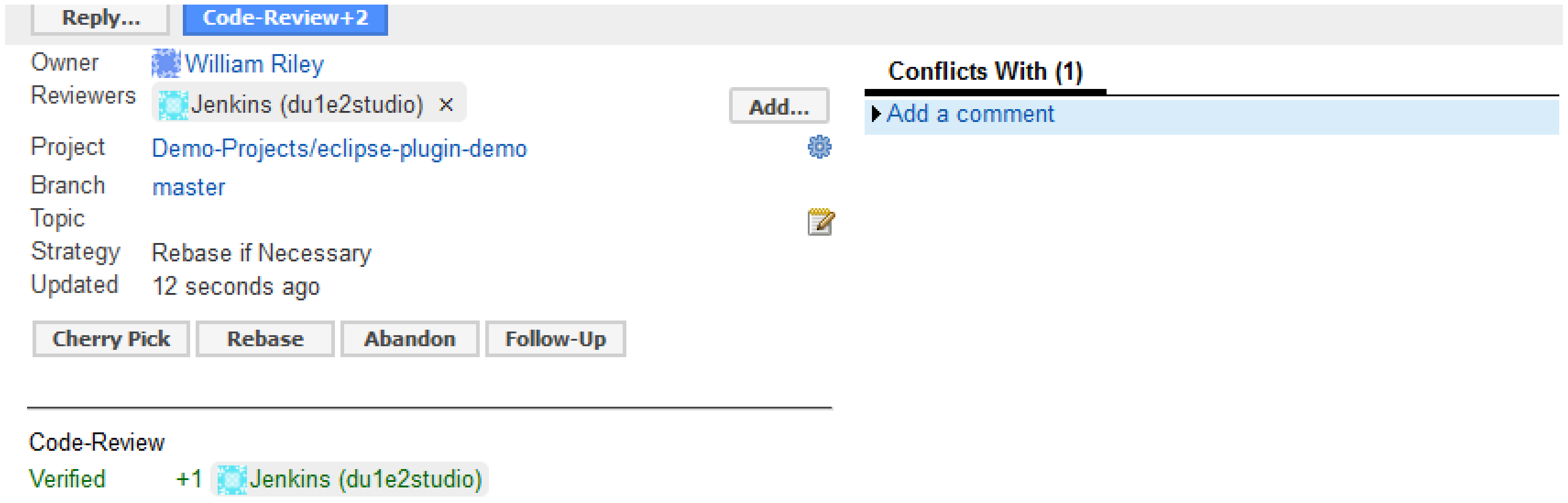
Owner	William Riley
Reviewers	Jenkins (du1e2stu
Project	Demo-Projects/eclips
Branch	demos/reeDemo
Topic	
Strategy	Rebase if Necessary
Updated	2 seconds ago

Code-Review	+2	William Riley
Verified	+1	Jenkins (du1e2studio)

Author	William Riley <william.riley@renesas.com>	Oct 20, 2015 15:52
Committer	William Riley <william.riley@renesas.com>	Oct 20, 2015 16:15

Merge Conflicts

- If there is a merge conflict preventing the submit the status will say “Merge Conflict”
- You will be warned during the review phase if there is a conflict with another pending change



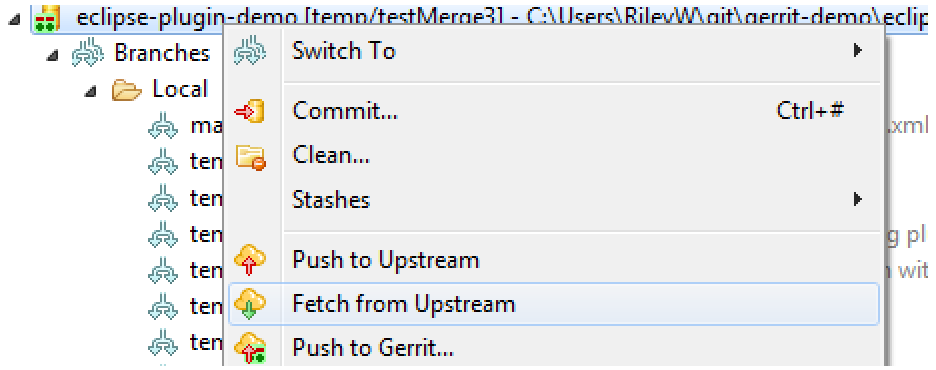
Owner	William Riley
Reviewers	Jenkins (du1e2studio) ×
Project	Demo-Projects/eclipse-plugin-demo
Branch	master
Topic	
Strategy	Rebase if Necessary
Updated	12 seconds ago

Code-Review
Verified +1 Jenkins (du1e2studio)

Merge Conflicts

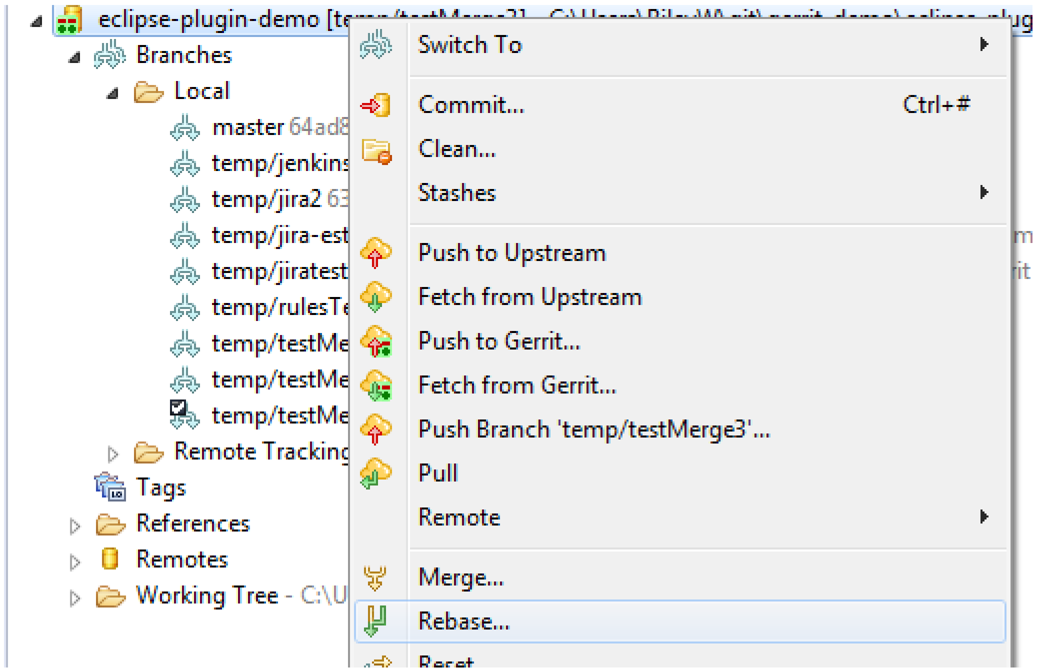
- To fix this you need to manually rebase the change onto the latest code

1. Switch to the branch you made the changes on
2. Updated your local repository with the latest changes using Fetch



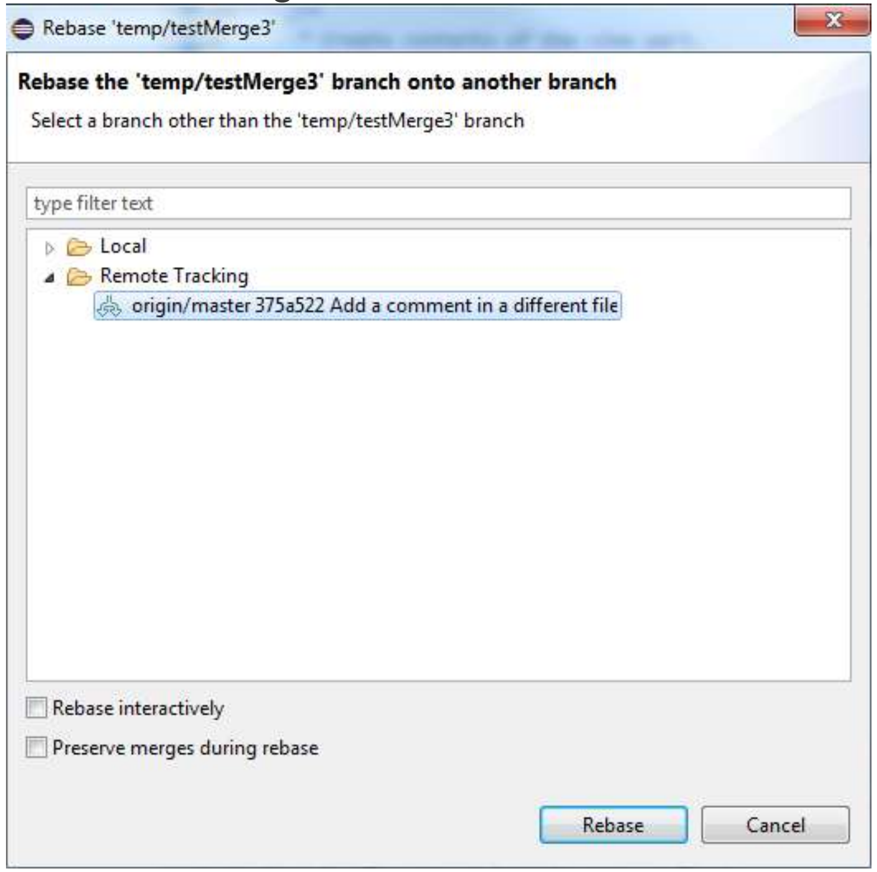
Merge Conflicts

1. Right click on the repository and select rebase



Merge Conflicts

- Select the branch under “Remote Tracking”



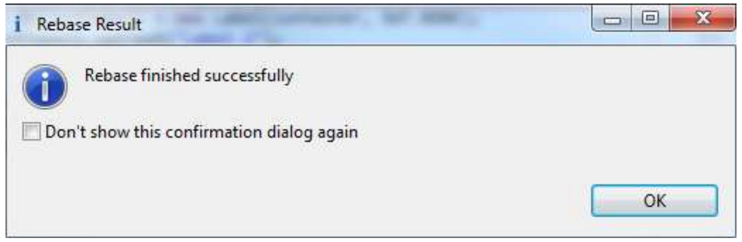
Rebase the 'temp/testMerge3' branch onto another branch

Select a branch other than the 'temp/testMerge3' branch

origin/master 375a522 Add a comment in a different file

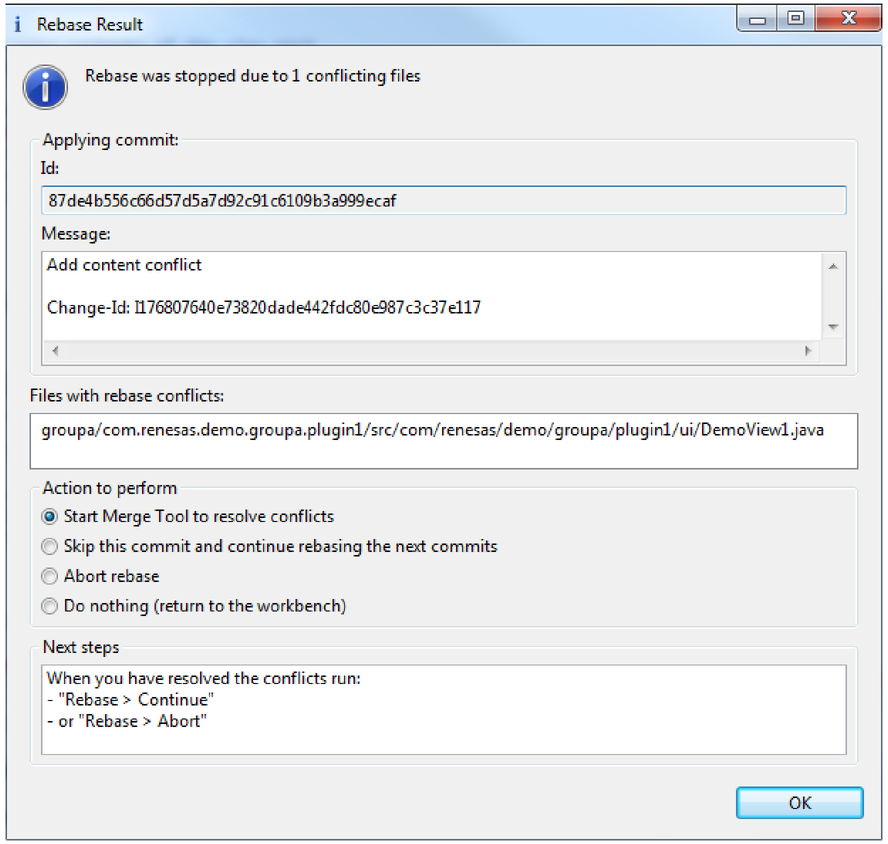
Rebasing

- Depending on the conflict you will see one of these 2 dialogs.
- If you see the one below EGit has merged the files automatically
- If you see the one on the right select “Start Merge Tool” and click “OK”


Rebase Result

Rebase finished successfully

☐ Don't show this confirmation dialog again


Rebase Result

Rebase was stopped due to 1 conflicting files

Applying commit:

Id:

87de4b556c66d57d5a7d92c91c6109b3a999ecaf

Message:

Add content conflict

Change-Id: I176807640e73820dade442fdc80e987c3c37e117

Files with rebase conflicts:

groupa/com.renesas.demo.groupa.plugin1/src/com/renesas/demo/groupa/plugin1/ui/DemoView1.java

Action to perform

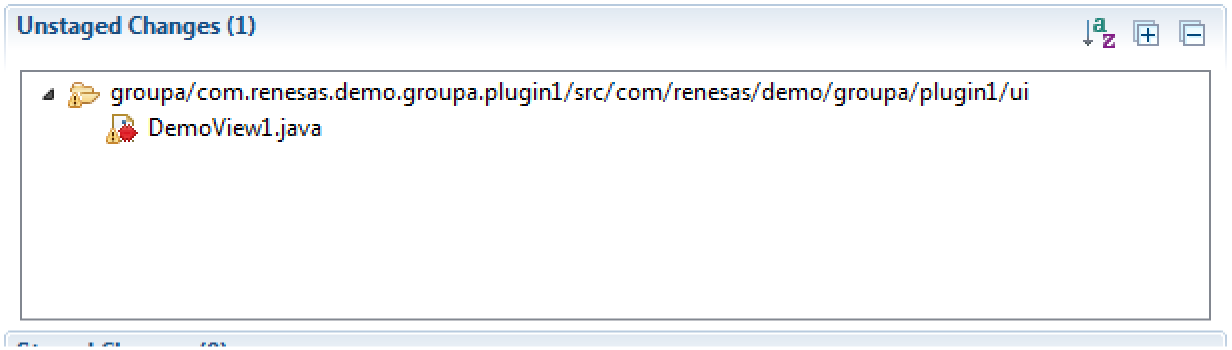
- ● Start Merge Tool to resolve conflicts
- ○ Skip this commit and continue rebasing the next commits
- ○ Abort rebase
- ○ Do nothing (return to the workbench)

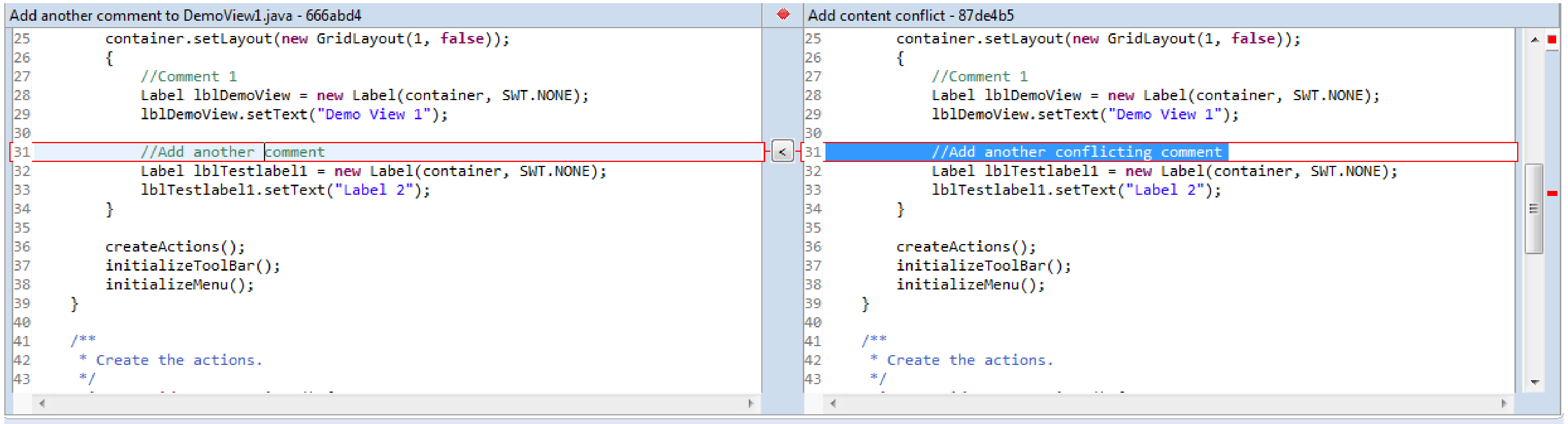
Next steps

When you have resolved the conflicts run:
- "Rebase > Continue"
- or "Rebase > Abort"

Rebasing (Conflicted)

- Conflicted files will show under “Unstaged Changes” with a red icon.


Unstaged Changes (1)
groupa/com.renesas.demo.groupa.plugin1/src/com/renesas/demo/groupa/plugin1/ui
DemoView1.java


Add another comment to DemoView1.java - 666abd4

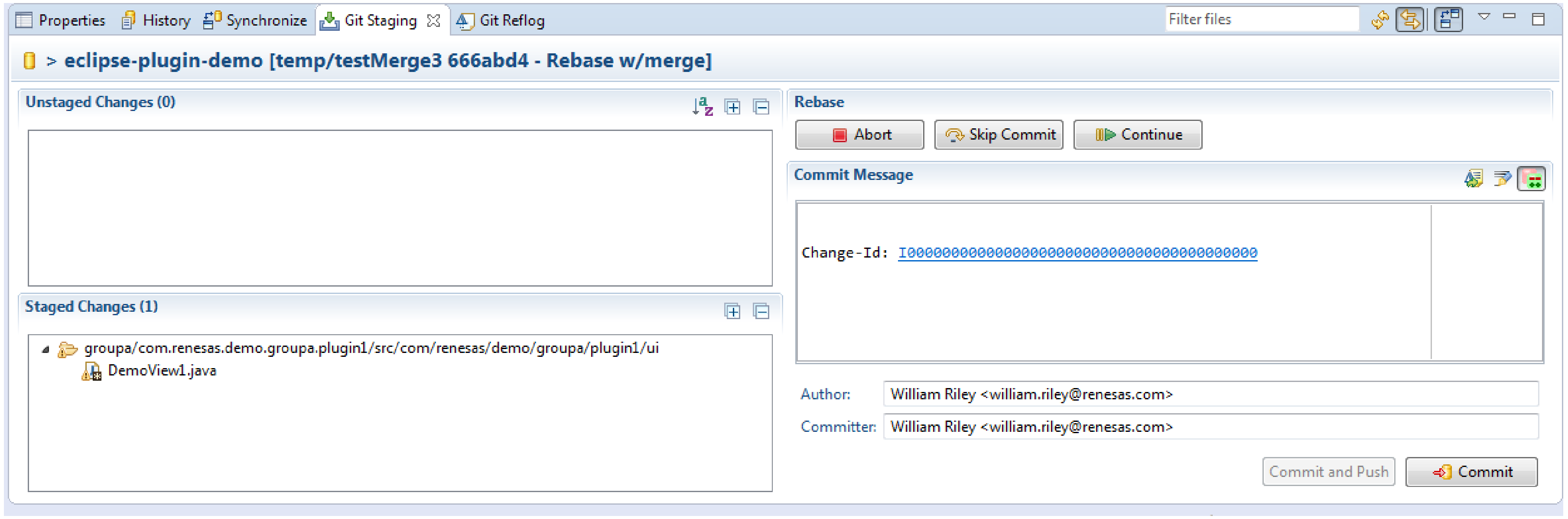
```
        container.setLayout(new GridLayout(1, false));
        {
            //Comment 1
            Label lblDemoView = new Label(container, SWT.NONE);
            lblDemoView.setText("Demo View 1");

            //Add another comment
            Label lblTestlabel1 = new Label(container, SWT.NONE);
            lblTestlabel1.setText("Label 2");
        }

        createActions();
        initializeToolBar();
        initializeMenu();
    }

    /**
     * Create the actions.
     */
```

Add content conflict - 87de4b5

```
        container.setLayout(new GridLayout(1, false));
        {
            //Comment 1
            Label lblDemoView = new Label(container, SWT.NONE);
            lblDemoView.setText("Demo View 1");

            //Add another conflicting comment
            Label lblTestlabel1 = new Label(container, SWT.NONE);
            lblTestlabel1.setText("Label 2");
        }

        createActions();
        initializeToolBar();
        initializeMenu();
    }

    /**
     * Create the actions.
     */
```

Rebasing (Conflicted)

- Stage the changed file
- When all conflicts resolved click “Continue”



> eclipse-plugin-demo [temp/testMerge3 666abd4 - Rebase w/merge]

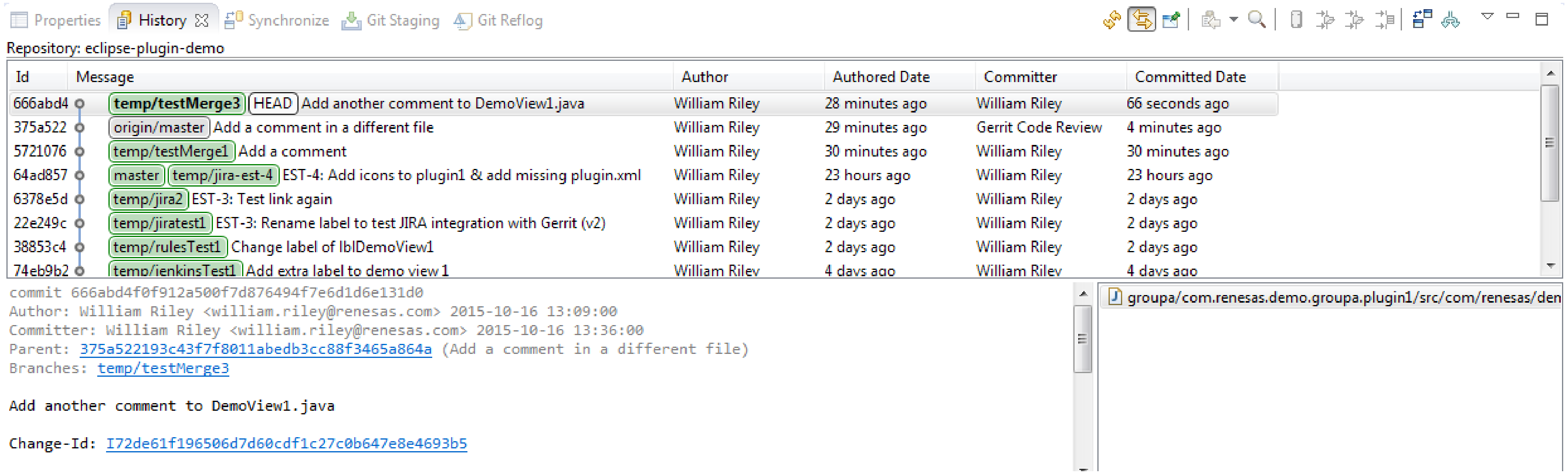
groupa/com.renesas.demo.groupa.plugin1/src/com/renesas/demo/groupa/plugin1/ui
DemoView1.java

Change-Id: I0000000000000000000000000000000000000000

Author: William Riley <william.riley@renesas.com>
Committer: William Riley <william.riley@renesas.com>

Submitting a change

- It is a good idea to check the commit to ensure the rebased change is correct
- Find it in the history view and double click the files to see changes



Repository: eclipse-plugin-demo

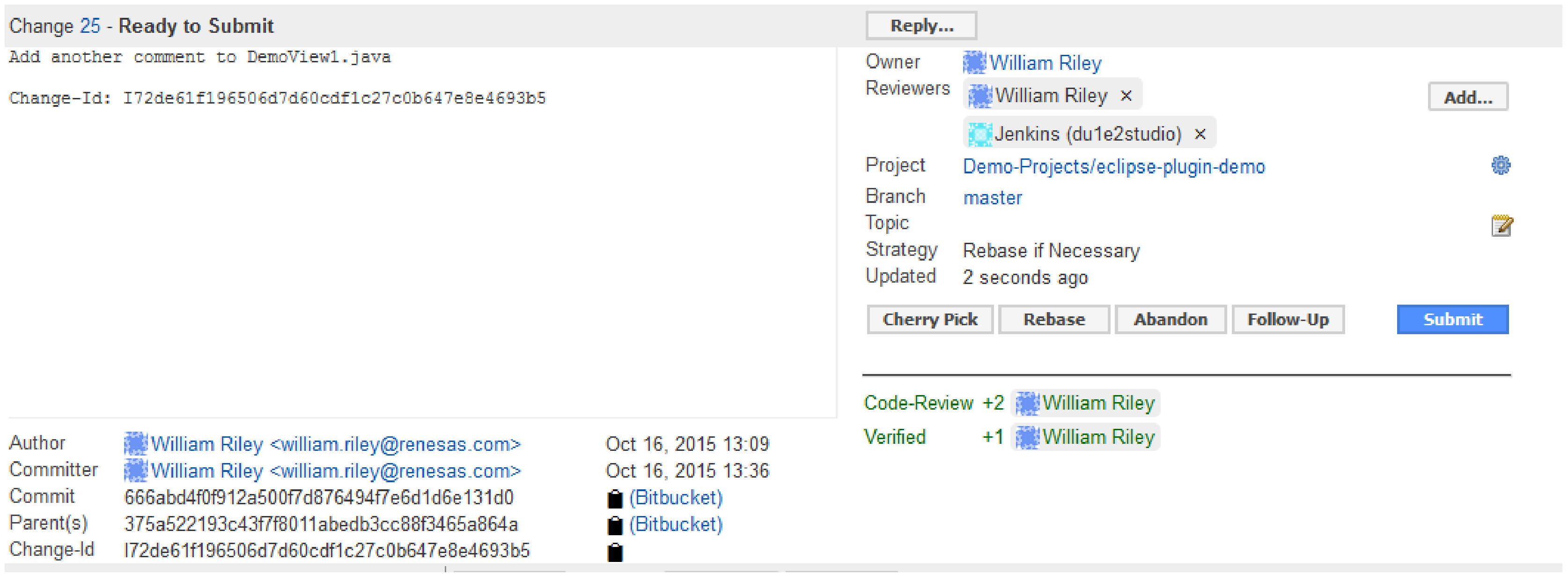
Id	Message	Author	Authored Date	Committer	Committed Date
666abd4	temp/testMerge3 HEAD Add another comment to DemoView1.java	William Riley	28 minutes ago	William Riley	66 seconds ago
375a522	origin/master Add a comment in a different file	William Riley	29 minutes ago	Gerrit Code Review	4 minutes ago
5721076	temp/testMerge1 Add a comment	William Riley	30 minutes ago	William Riley	30 minutes ago
64ad857	master temp/jira-est-4 EST-4: Add icons to plugin1 & add missing plugin.xml	William Riley	23 hours ago	William Riley	23 hours ago
6378e5d	temp/jira2 EST-3: Test link again	William Riley	2 days ago	William Riley	2 days ago
22e249c	temp/jiratest1 EST-3: Rename label to test JIRA integration with Gerrit (v2)	William Riley	2 days ago	William Riley	2 days ago
38853c4	temp/rulesTest1 Change label of lblDemoView1	William Riley	2 days ago	William Riley	2 days ago
74eb9b2	temp/jenkinsTest1 Add extra label to demo view 1	William Riley	4 days ago	William Riley	4 days ago

```
commit 666abd4f0f912a500f7d876494f7e6d1d6e131d0
Author: William Riley <william.riley@renesas.com> 2015-10-16 13:09:00
Committer: William Riley <william.riley@renesas.com> 2015-10-16 13:36:00
Parent: 375a522193c43f7f8011abedb3cc88f3465a864a (Add a comment in a different file)
Branches: temp/testMerge3

Add another comment to DemoView1.java

Change-Id: I72de61f196506d7d60cdf1c27c0b647e8e4693b5
```

groupa/com.renesas.demo.groupa.plugin1/src/com/renesas/den

Submitting a change

- If it worked the change will now say “Ready to Submit”
- The responsible submitter should now click “Submit”
- If the code changed during the rebase then the review process will need to be completed again



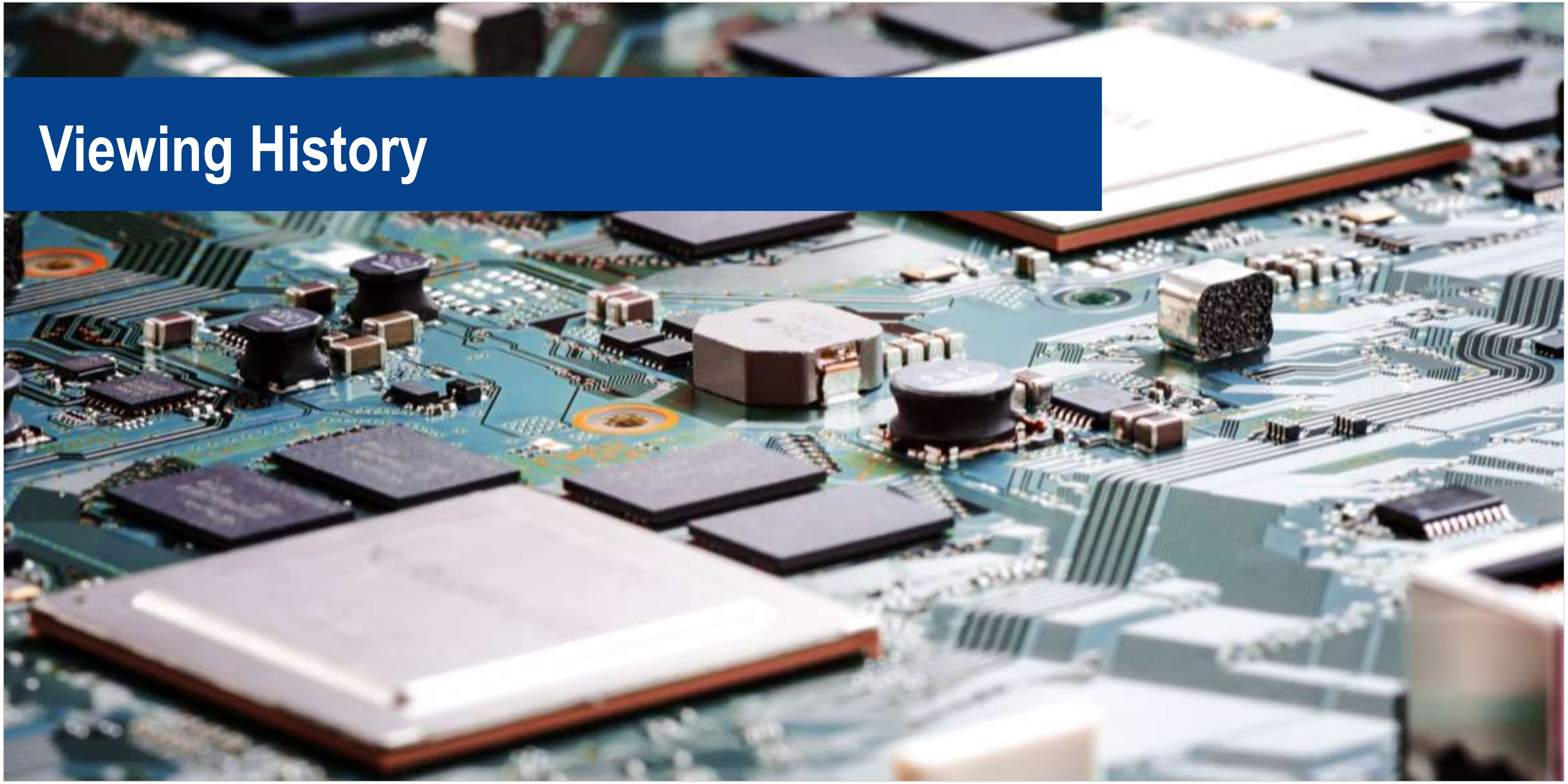
Change 25 - **Ready to Submit**

```
Add another comment to DemoView1.java

Change-Id: I72de61f196506d7d60cdf1c27c0b647e8e4693b5
```

Owner	William Riley
Reviewers	William Riley × Jenkins (du1e2studio) ×
Project	Demo-Projects/eclipse-plugin-demo
Branch	master
Topic	
Strategy	Rebase if Necessary
Updated	2 seconds ago

Code-Review	+2	William Riley
Verified	+1	William Riley

Author	William Riley <william.riley@renesas.com>	Oct 16, 2015 13:09
Committer	William Riley <william.riley@renesas.com>	Oct 16, 2015 13:36
Commit	666abd4f0f912a500f7d876494f7e6d1d6e131d0	(Bitbucket)
Parent(s)	375a522193c43f7f8011abedb3cc88f3465a864a	(Bitbucket)
Change-Id	I72de61f196506d7d60cdf1c27c0b647e8e4693b5	



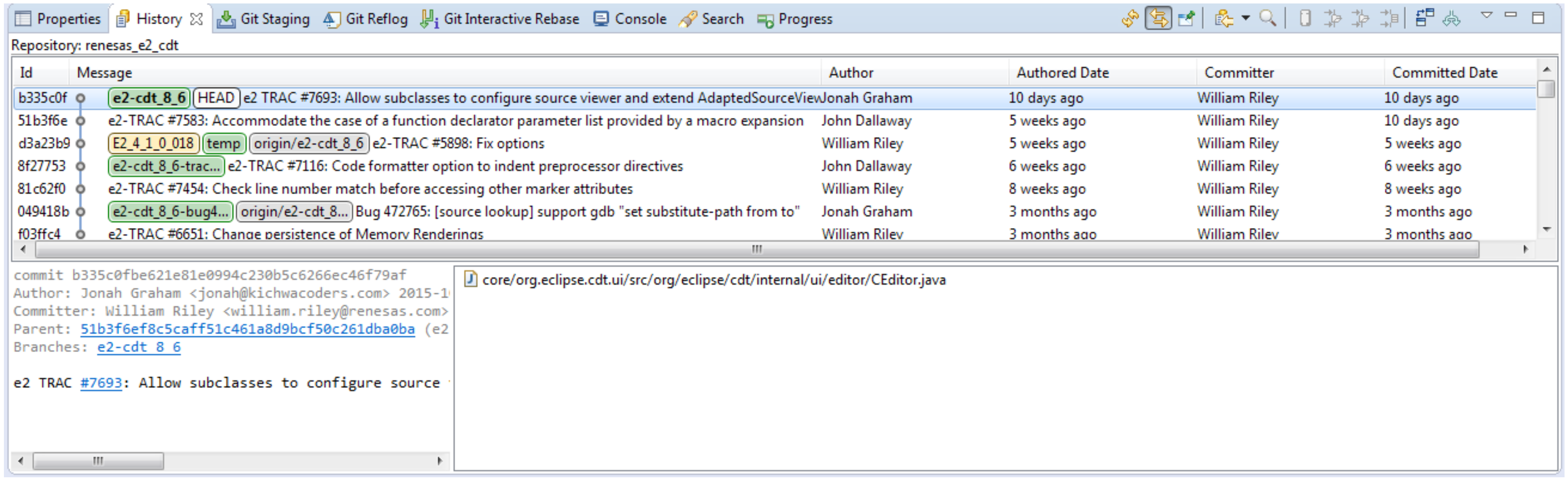
Viewing History

How?

- At some point you are going to want to view this history, either of the whole repository or an individual file or directory.
- Can be done within EGit or though Bitbucket

EGit History View

- The History view in Egit allows you to see the history of the whole repository or specific files



Repository: renesas_e2_cdt

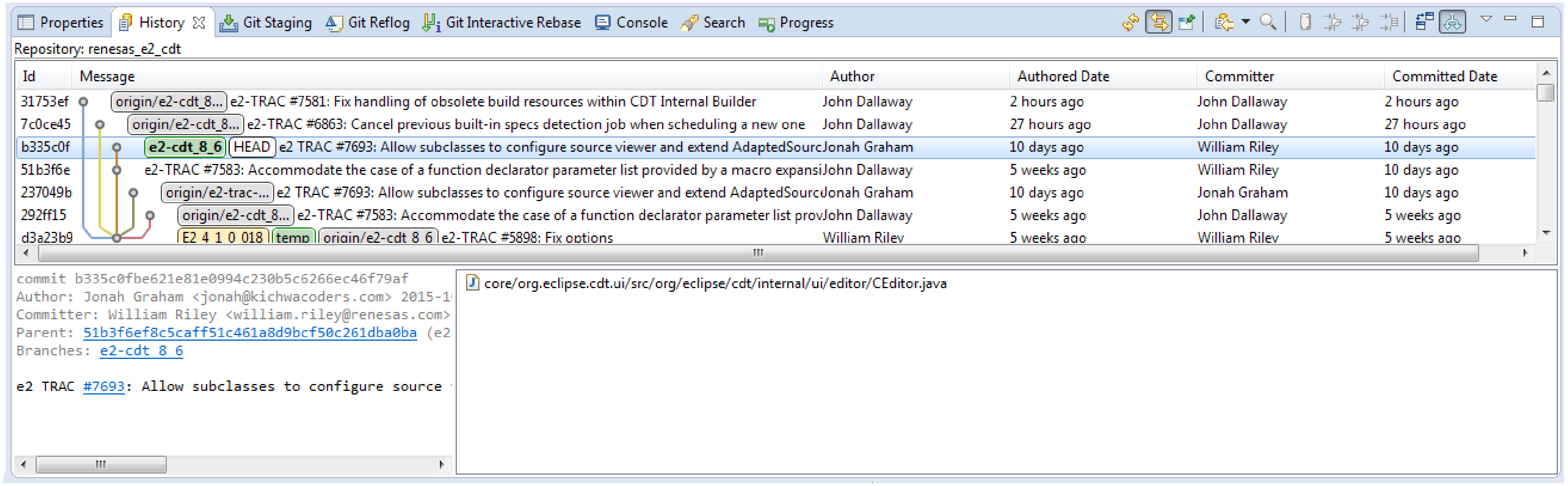
Id	Message	Author	Authored Date	Committer	Committed Date
b335c0f	e2-cdt_8_6 HEAD e2 TRAC #7693: Allow subclasses to configure source viewer and extend AdaptedSourceView	Jonah Graham	10 days ago	William Riley	10 days ago
51b3f6e	e2-TRAC #7583: Accommodate the case of a function declarator parameter list provided by a macro expansion	John Dallaway	5 weeks ago	William Riley	10 days ago
d3a23b9	E2_4_1_0_018 temp origin/e2-cdt_8_6 e2-TRAC #5898: Fix options	William Riley	5 weeks ago	William Riley	5 weeks ago
8f27753	e2-cdt_8_6-trac... e2-TRAC #7116: Code formatter option to indent preprocessor directives	John Dallaway	6 weeks ago	William Riley	6 weeks ago
81c62f0	e2-TRAC #7454: Check line number match before accessing other marker attributes	William Riley	8 weeks ago	William Riley	8 weeks ago
049418b	e2-cdt_8_6-bug4... origin/e2-cdt_8... Bug 472765: [source lookup] support gdb "set substitute-path from to"	Jonah Graham	3 months ago	William Riley	3 months ago
f03ffc4	e2-TRAC #6651: Change persistence of Memory Renderings	William Riley	3 months ago	William Riley	3 months ago

```
commit b335c0fbe621e81e0994c230b5c6266ec46f79af
Author: Jonah Graham <jonah@kichwacoders.com> 2015-1
Committer: William Riley <william.riley@renesas.com>
Parent: 51b3f6ef8c5caff51c461a8d9bcf50c261dba0ba (e2
Branches: e2-cdt 8 6

e2 TRAC #7693: Allow subclasses to configure source
```

core/org.eclipse.cdt.ui/src/org/eclipse/cdt/internal/ui/editor/CEditor.java

EGit History View (All branches)

- Can view history for all branches at the same time
- Commits are ordered by commit date



Repository: renesas_e2_cdt

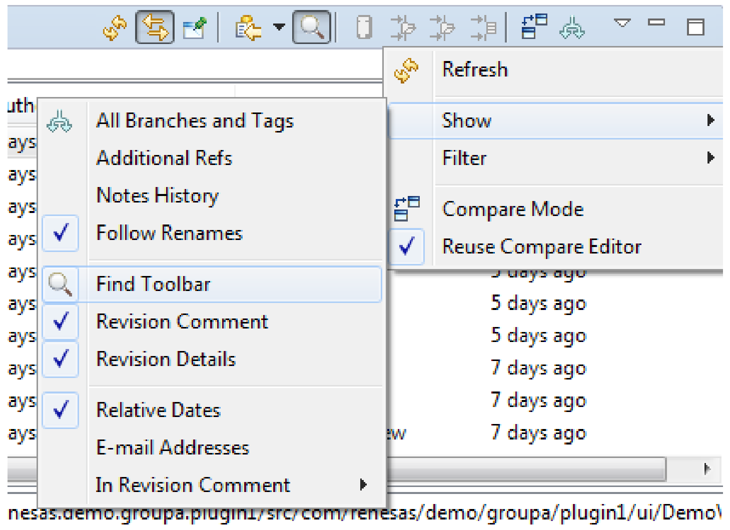
Id	Message	Author	Authored Date	Committer	Committed Date
31753ef	origin/e2-cdt_8... e2-TRAC #7581: Fix handling of obsolete build resources within CDT Internal Builder	John Dallaway	2 hours ago	John Dallaway	2 hours ago
7c0ce45	origin/e2-cdt_8... e2-TRAC #6863: Cancel previous built-in specs detection job when scheduling a new one	John Dallaway	27 hours ago	John Dallaway	27 hours ago
b335c0f	e2-cdt_8_6 HEAD e2 TRAC #7693: Allow subclasses to configure source viewer and extend AdaptedSourc	Jonah Graham	10 days ago	William Riley	10 days ago
51b3f6e	e2-TRAC #7583: Accommodate the case of a function declarator parameter list provided by a macro expansi	John Dallaway	5 weeks ago	William Riley	10 days ago
237049b	origin/e2-trac-... e2 TRAC #7693: Allow subclasses to configure source viewer and extend AdaptedSourc	Jonah Graham	10 days ago	Jonah Graham	10 days ago
292ff15	origin/e2-cdt_8... e2-TRAC #7583: Accommodate the case of a function declarator parameter list prov	John Dallaway	5 weeks ago	John Dallaway	5 weeks ago
d3a23b9	E2_4_1_0_018 temp origin/e2-cdt_8_6 e2-TRAC #5898: Fix options	William Riley	5 weeks ago	William Riley	5 weeks ago

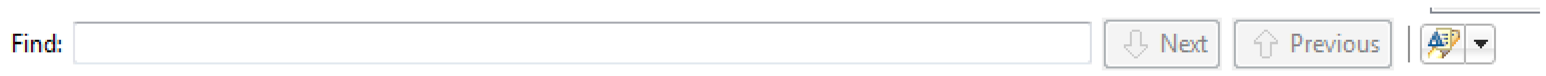
```
commit b335c0fbe621e81e0994c230b5c6266ec46f79af
Author: Jonah Graham <jonah@kichwacoders.com> 2015-1
Committer: William Riley <william.riley@renesas.com>
Parent: 51b3f6ef8c5caff51c461a8d9bcf50c261dba0ba (e2
Branches: e2-cdt_8_6

e2 TRAC #7693: Allow subclasses to configure source
```

core/org.eclipse.cdt.ui/src/org/eclipse/cdt/internal/ui/editor/CEditor.java

Egit History View (Search)

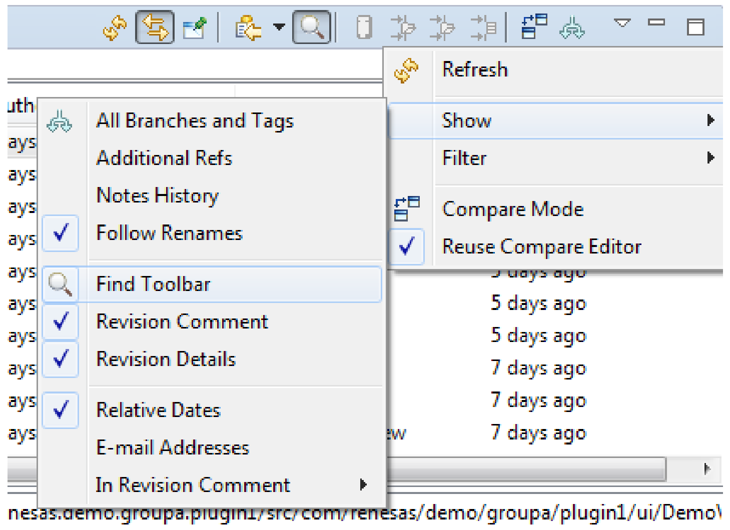
- You can search the git history from the History view
- In the view menu go to "Show -> Find Toolbar"
- A toolbar will be added to the bottom of the view
- By default it will search all fields

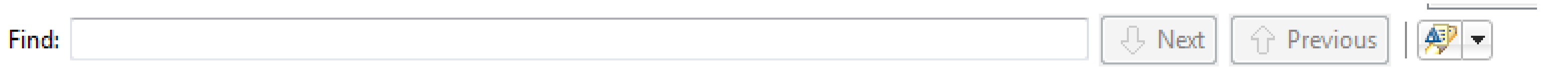




Search

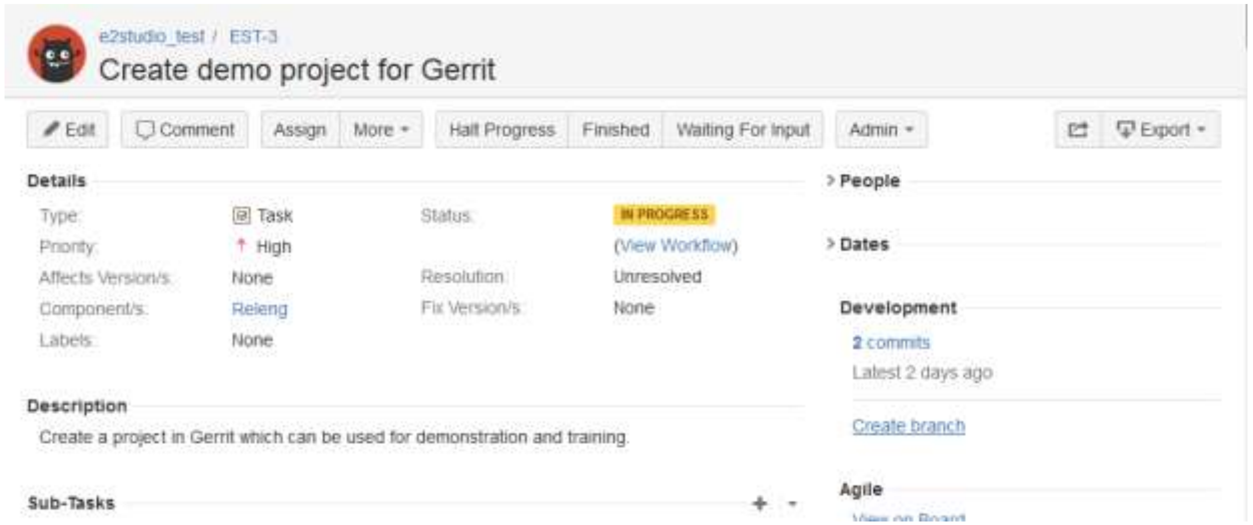
- You can search the git history from the History view
- In the view menu go to "Show -> Find Toolbar"
- A toolbar will be added to the bottom of the view
- By default it will search all fields





From JIRA Issue

- If you have bitbucket access you will be able to see all commits related to a JIRA issue on the JIRA page
- Look for the “Development” pane on the right
- Click “n commits”


e2studio_test / EST-3

Create demo project for Gerrit

Details

Type:	Task	Status:	IN PROGRESS
Priority:	High		(View Workflow)
Affects Version/s:	None	Resolution:	Unresolved
Component/s:	Releng	Fix Version/s:	None
Labels:	None		

Description

Create a project in Gerrit which can be used for demonstration and training.

Sub-Tasks

People

Dates

Development

2 commits

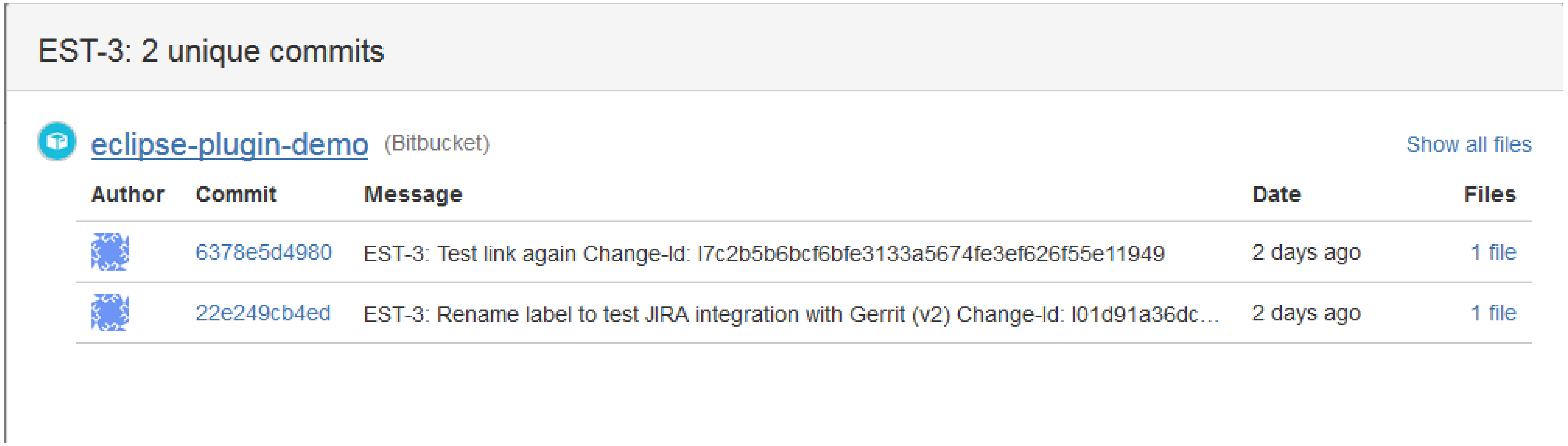
Latest 2 days ago

Create branch

Agile

From JIRA Issue

- A list of all commits referencing that JIRA issue will be displayed
- Click on the commit ID to be taken to that commit in Bitbucket
- The 1st time you do this you may see a message asking you to authorise JIRA to access Bitbucket



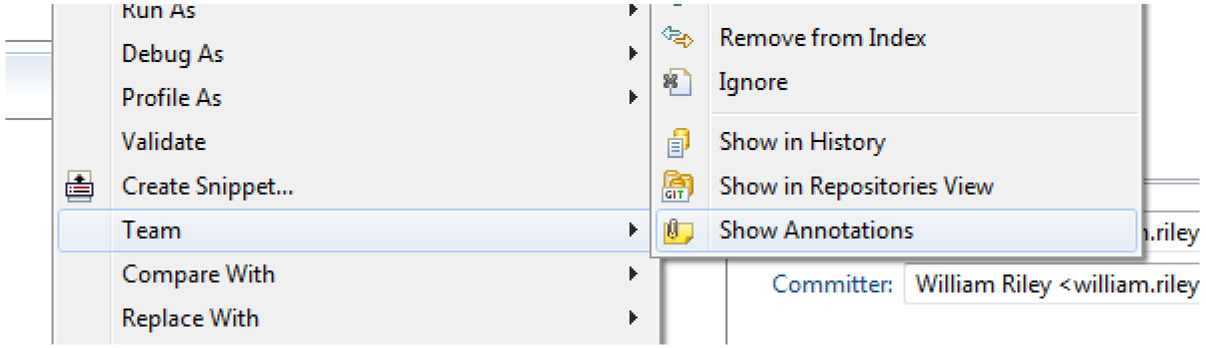
EST-3: 2 unique commits

eclipse-plugin-demo (Bitbucket) Show all files

Author	Commit	Message	Date	Files
	6378e5d4980	EST-3: Test link again Change-Id: I7c2b5b6bcf6bfe3133a5674fe3ef626f55e11949	2 days ago	1 file
	22e249cb4ed	EST-3: Rename label to test JIRA integration with Gerrit (v2) Change-Id: I01d91a36dc…	2 days ago	1 file

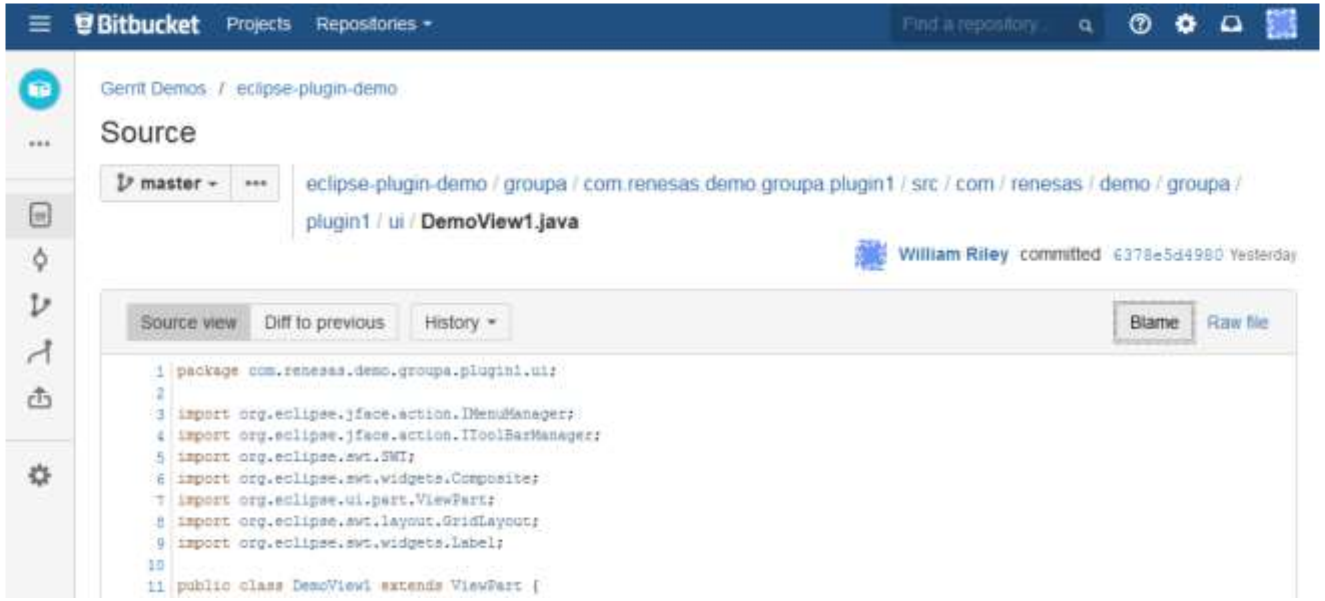
Blame Annotations (Egit)

- Open the file in Eclipse
- Right click in the file do Team -> Show Annotations



Blame Annotations (Bitbucket)

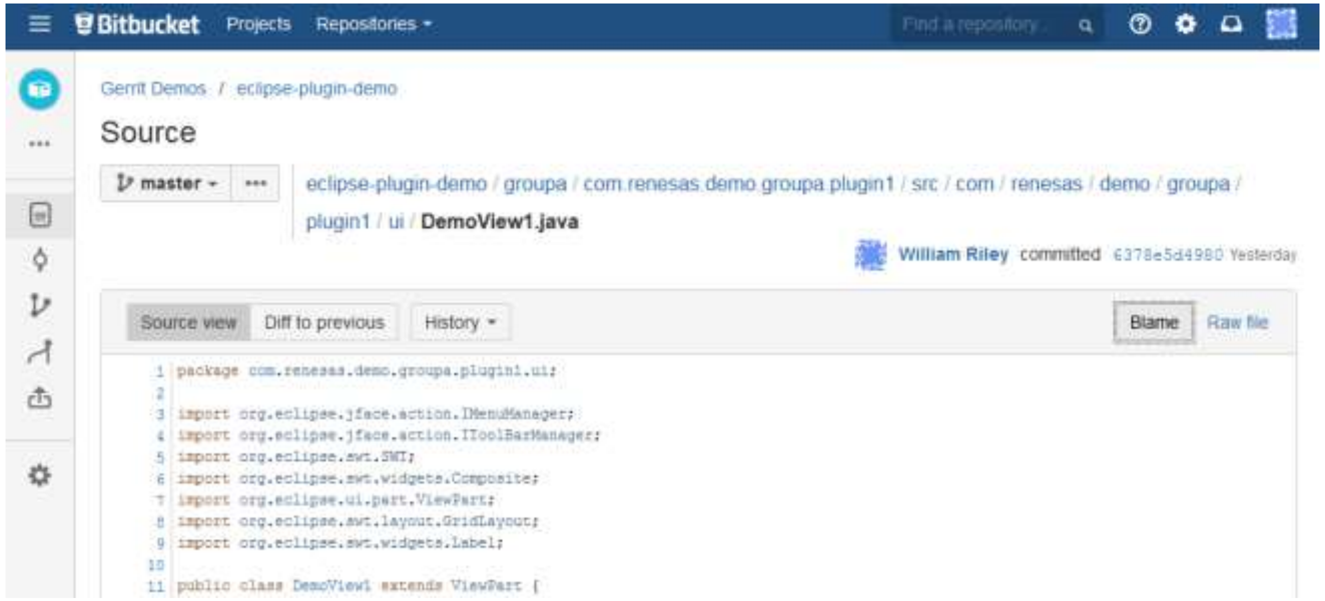
- Open the file in in Bitbucket
- Click Blame


Bitbucket
Projects
Repositories
Find a repository...
Gerrit Demos / eclipse-plugin-demo
Source
master
eclipse-plugin-demo / groupa / com.renesas.demo.groupa.plugin1 / src / com / renesas / demo / groupa / plugin1 / ui / DemoView1.java
William Riley committed 6378e5d4980 Yesterday
Source view
Diff to previous
History
Blame
Raw file

```
package com.renesas.demo.groupa.plugin1.ui;

import org.eclipse.jface.action.IMenuManager;
import org.eclipse.jface.action.IToolBarManager;
import org.eclipse.swt.SWT;
import org.eclipse.swt.widgets.Composite;
import org.eclipse.ui.part.ViewPart;
import org.eclipse.swt.layout.GridLayout;
import org.eclipse.swt.widgets.Label;

public class DemoView1 extends ViewPart {
```

Blame Annotations (Bitbucket)

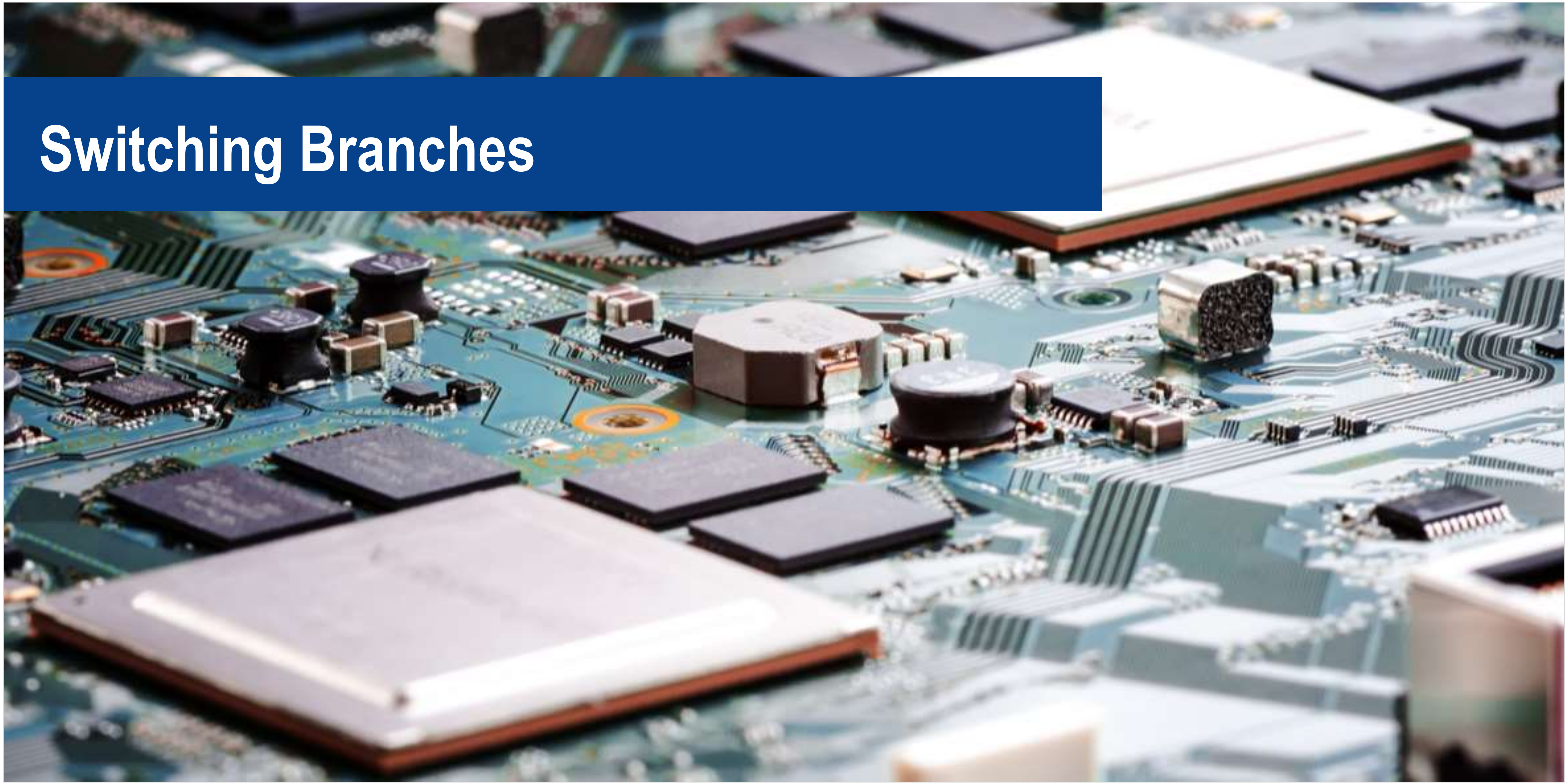
- Open the file in in Bitbucket
- Click Blame



eclipse-plugin-demo / groupa / com.renesas.demo.groupa.plugin1 / src / com / renesas / demo / groupa / plugin1 / ui / DemoView1.java

William Riley committed 6378e5d4980 Yesterday

```
package com.renesas.demo.groupa.plugin1.ui;

import org.eclipse.jface.action.IMenuManager;
import org.eclipse.jface.action.IToolBarManager;
import org.eclipse.swt.SWT;
import org.eclipse.swt.widgets.Composite;
import org.eclipse.ui.part.ViewPart;
import org.eclipse.swt.layout.GridLayout;
import org.eclipse.swt.widgets.Label;

public class DemoView1 extends ViewPart {
```



Switching Branches

To existing local

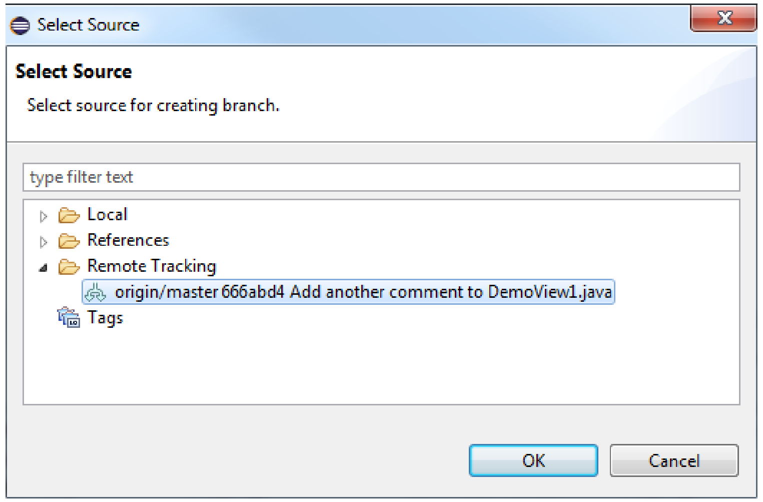
1. Select the repository in the Git Repositories view
2. Right click on the repository
3. Go to “Switch To…”
4. Select the branch
5. Optionally if this branch was created from a remote branch do a pull to update to the latest version

To new local

1. Select the repository in the Git Repositories view
2. Right click on the repository
3. Go to “Switch To…”
4. New Branch
5. Next to “Source” click Select
6. Browse to Local and select the branch you want
7. Enter your branch name
8. Select “Rebase…” under “When doing a pull”
9. Click Finish

To new local from remote

1. Do a fetch to ensure you have the latest remote branches and commits
2. Go to "Switch To…"
3. New Branch
4. Next to "Source" click Select
5. Browse to remote tracking and select the branch you want
6. Enter your branch name
7. Click Finish



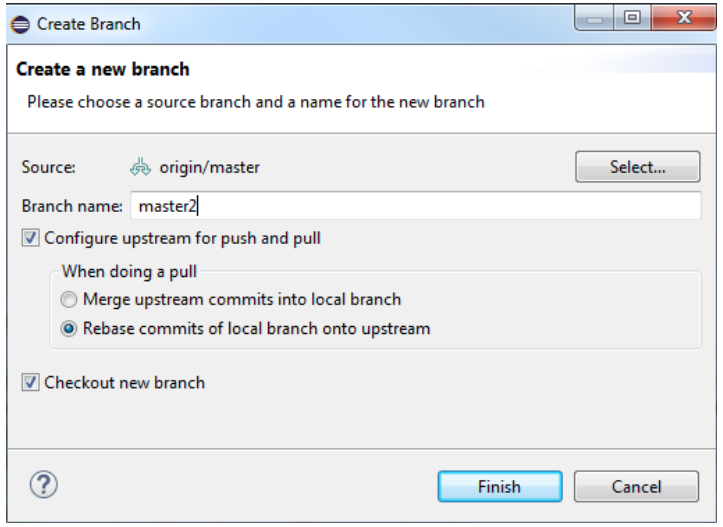
Select Source

Select Source

Select source for creating branch.

type filter text

- Local
- References
- Remote Tracking
  - origin/master 666abd4 Add another comment to DemoView1.java
- Tags



Create Branch

Create a new branch

Please choose a source branch and a name for the new branch

Source: origin/master

Branch name: master2

[✓] Configure upstream for push and pull

When doing a pull

( ) Merge upstream commits into local branch

(●) Rebase commits of local branch onto upstream

[✓] Checkout new branch

Applying a change to another branch

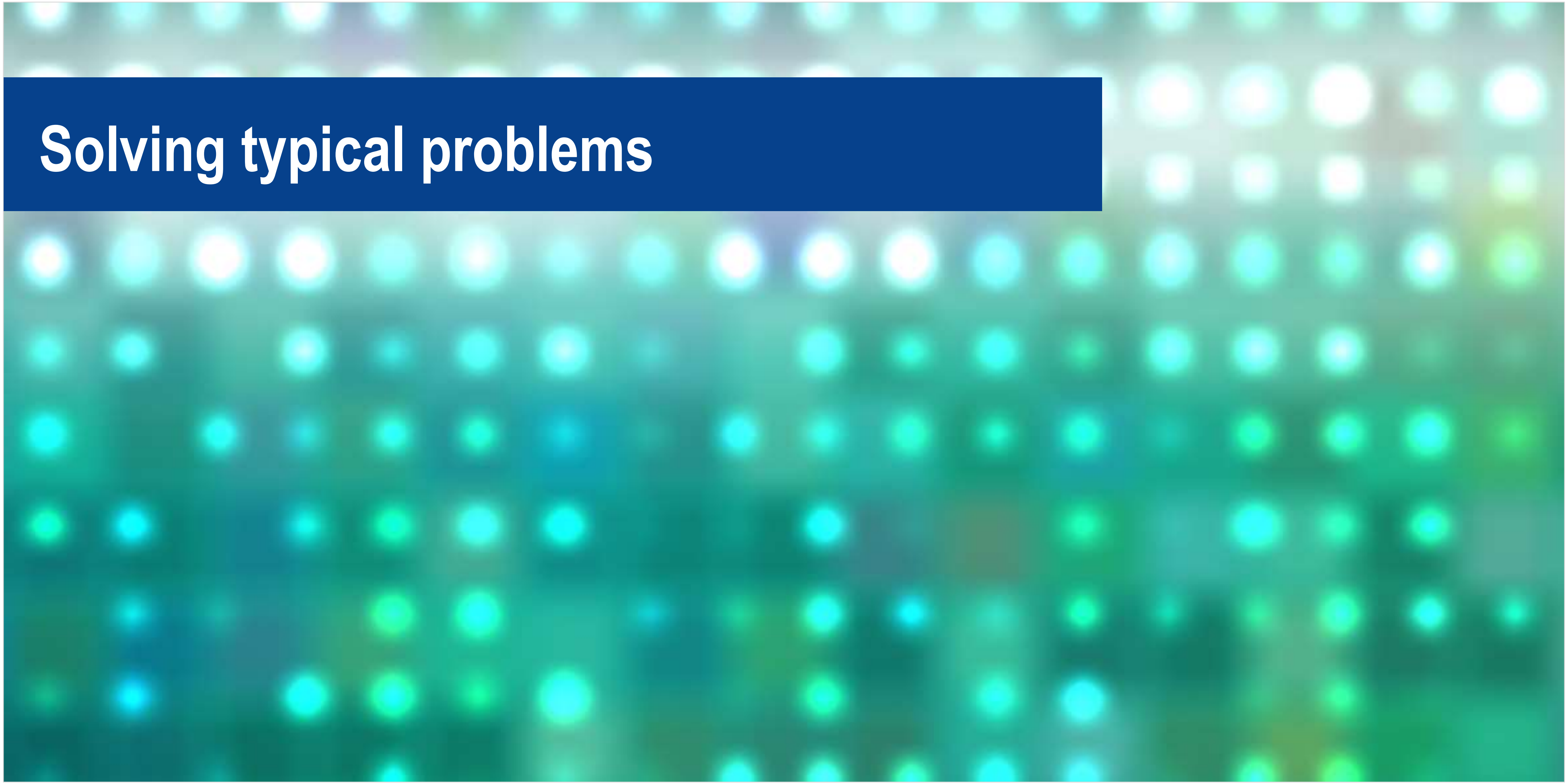
Applying a change to a different branch

- Demo

Drafts

Drafts

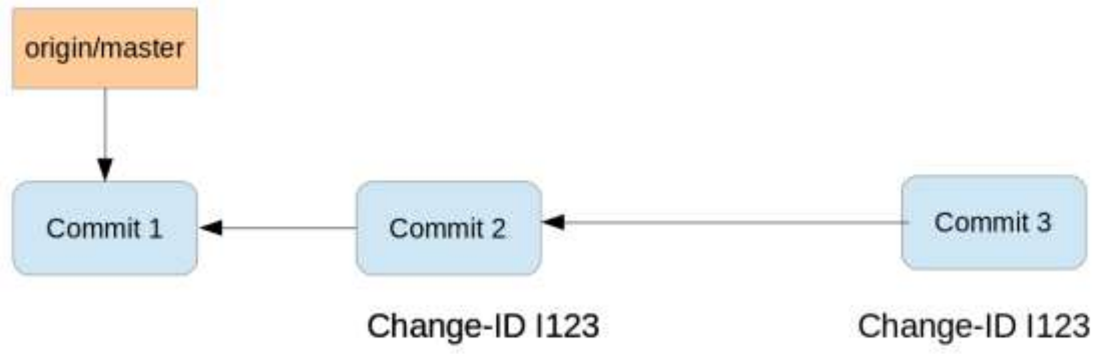
- You Can push a change to Gerrit as a Draft
- Draft changes are only visible to Project Owners, the change owner and reviewers
- Can be converted into a normal change by publishing
- Or deleted completely.

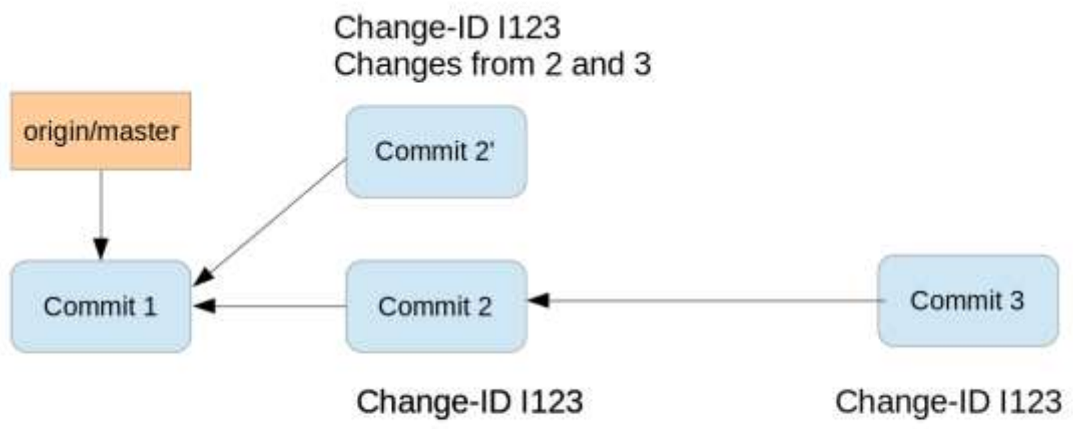


Solving typical problems

You forgot to amend the commit

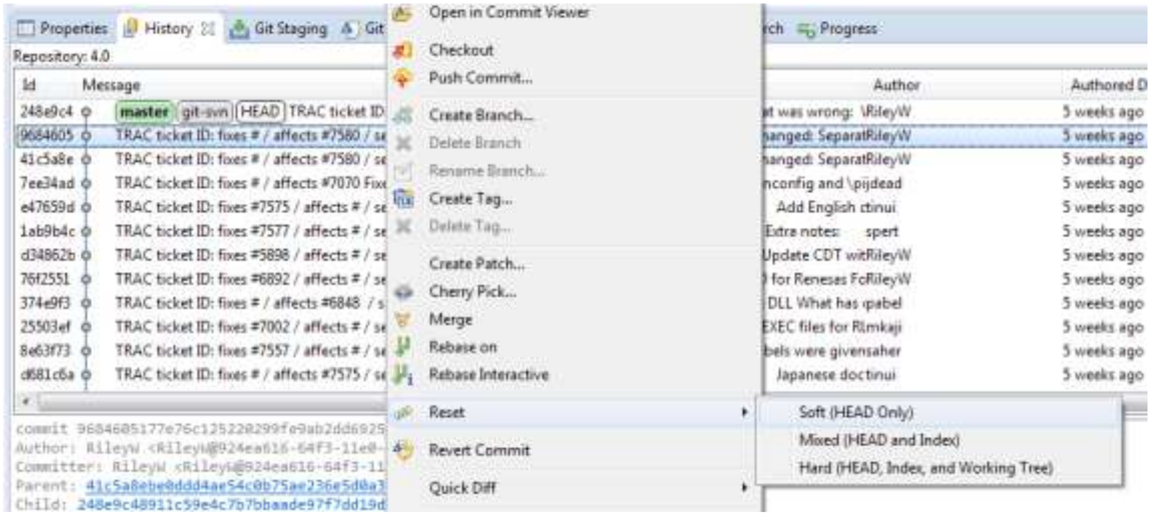
This is either if you you try to push a commit which contains the same ChangeId as a predecessor commit Gerrit will reject it and responds with the error message: "squash commits first". Or you forget to amend and notice before trying to push.





You forgot to amend the commit

- In this case you need to squash the commits. This results in a new commit which can be pushed to Gerrit. As you use the same Change-ID this pushed commit will update the Gerrit review.
- An easy solution to handle this is to do a soft reset in Git to the previous commit and commit the change files again, this time with the amend option.
- To do this open the history, select the commit before the extra one, right click and go to Reset -> Soft



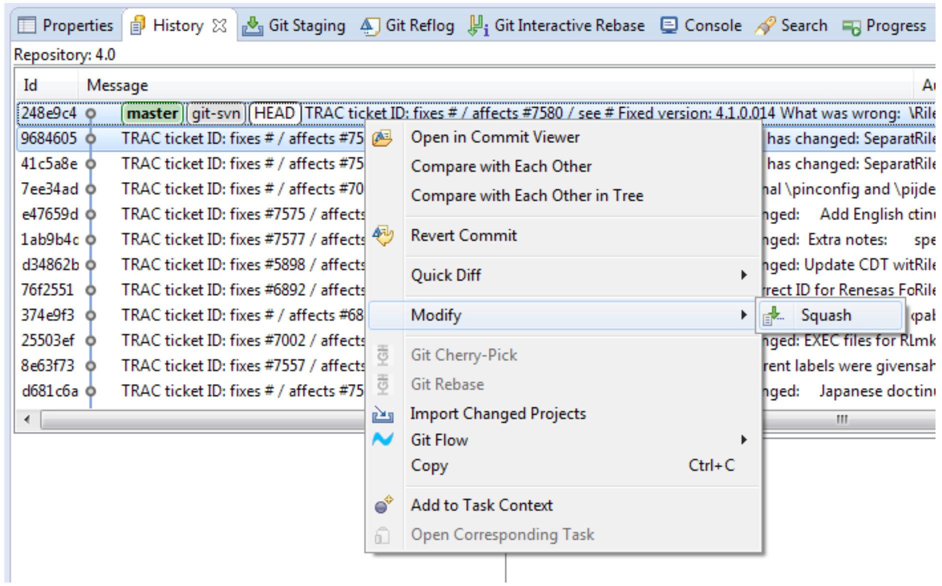
Repository: 4.0

Id	Message		Author	Authored D
248e9c4	master git-svn HEAD TRAC ticket ID	at was wrong:	\RileyW	5 weeks ago
9684605	TRAC ticket ID: fixes # / affects #7580 / se	hanged: Separat	RileyW	5 weeks ago
41c5a8e	TRAC ticket ID: fixes # / affects #7580 / se	hanged: Separat	RileyW	5 weeks ago
7ee34ad	TRAC ticket ID: fixes # / affects #7070 Fixe	nconfig and	\pijdead	5 weeks ago
e47659d	TRAC ticket ID: fixes #7575 / affects # / se	Add English	ctinui	5 weeks ago
1ab9b4c	TRAC ticket ID: fixes #7577 / affects # / se	Extra notes:	spert	5 weeks ago
d34862b	TRAC ticket ID: fixes #5898 / affects # / se	Update CDT wit	RileyW	5 weeks ago
76f2551	TRAC ticket ID: fixes #6892 / affects # / se	for Reneses Fc	RileyW	5 weeks ago
374e9f3	TRAC ticket ID: fixes # / affects #6848 / s	DLL What has	ıpabel	5 weeks ago
25503ef	TRAC ticket ID: fixes #7002 / affects # / se	EXEC files for	Rlmkaji	5 weeks ago
8e63f73	TRAC ticket ID: fixes #7557 / affects # / se	bels were given	saher	5 weeks ago
d681c6a	TRAC ticket ID: fixes # / affects #7575 / se	Japanese doc	tinui	5 weeks ago

```
commit 9684605177e76c125220299fe9ab2dd6925
Author: RileyW <RileyW@924ea616-64f3-11e0-
Committer: RileyW <RileyW@924ea616-64f3-11
Parent: 41c5a8ebe0ddd4ae54c0b75ae236e5d0a3
Child: 248e9c48911c59e4c7b7bbaade97f7dd19d
```

You forgot to amend the commit

Alternatively select the 2 commits in history view, and select Modify -> Squash



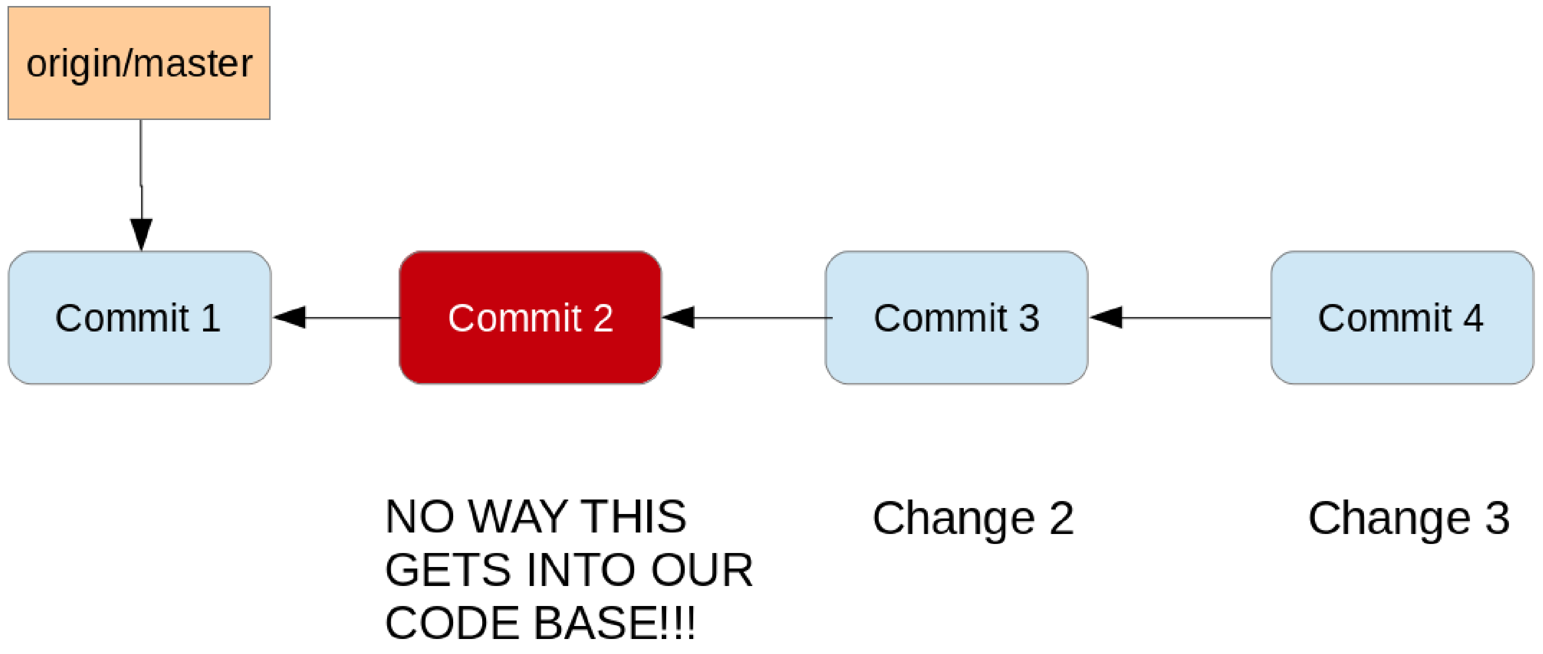
Repository: 4.0

Id	Message	A
248e9c4	master git-svn HEAD TRAC ticket ID: fixes # / affects #7580 / see # Fixed version: 4.1.0.014 What was wrong: \Rile	
9684605	TRAC ticket ID: fixes # / affects #75 … has changed: SeparatRile	
41c5a8e	TRAC ticket ID: fixes # / affects #75 … has changed: SeparatRile	
7ee34ad	TRAC ticket ID: fixes # / affects #70 … hal \pinconfig and \pijde	
e47659d	TRAC ticket ID: fixes #7575 / affects … nged: Add English ctin	
1ab9b4c	TRAC ticket ID: fixes #7577 / affects … nged: Extra notes: spe	
d34862b	TRAC ticket ID: fixes #5898 / affects … nged: Update CDT witRile	
76f2551	TRAC ticket ID: fixes #6892 / affects … rrect ID for Renesas FoRile	
374e9f3	TRAC ticket ID: fixes # / affects #68 … ‹pal	
25503ef	TRAC ticket ID: fixes #7002 / affects … nged: EXEC files for RLmk	
8e63f73	TRAC ticket ID: fixes #7557 / affects … rent labels were givensah	
d681c6a	TRAC ticket ID: fixes # / affects #75 … nged: Japanese doctin	

Remove a bad commit from a series of commits

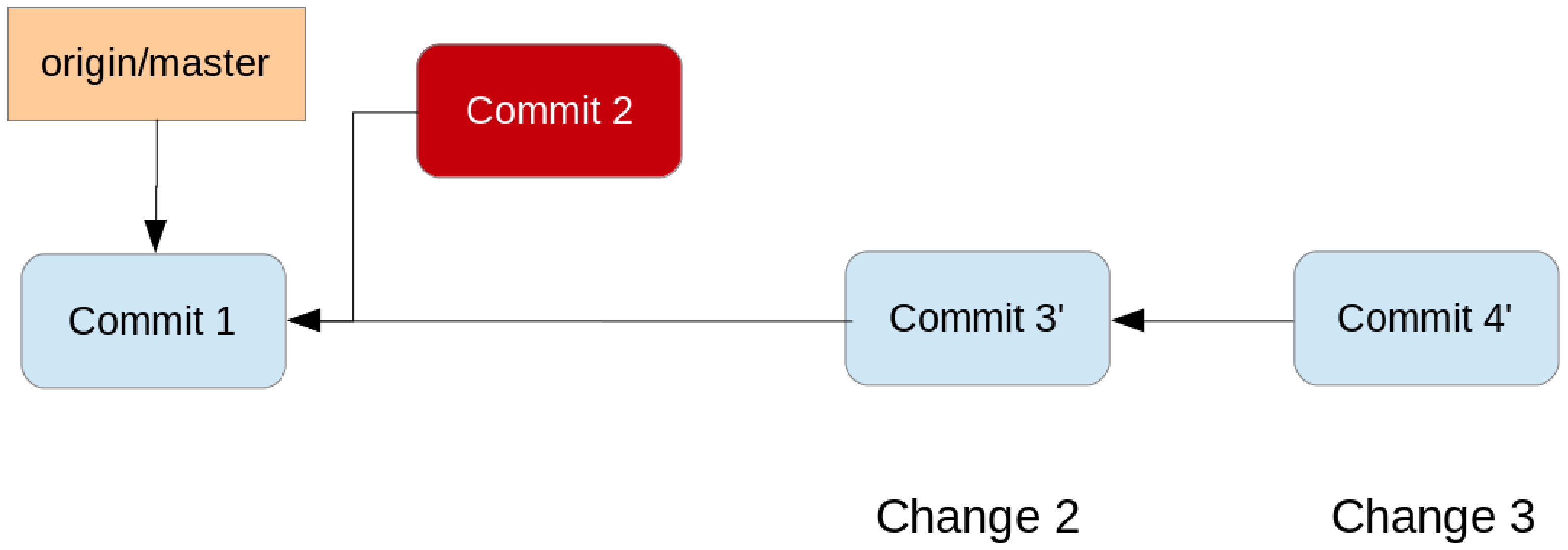
If you create a series of dependent commits, it might happen that one of the commits is rejected during the review process.

In this case you cannot merge the other commits as they still have a dependency to the "bad" commit.



Remove a bad commit from a series of commits

The solution is to use interactive rebase to skip the bad commit.



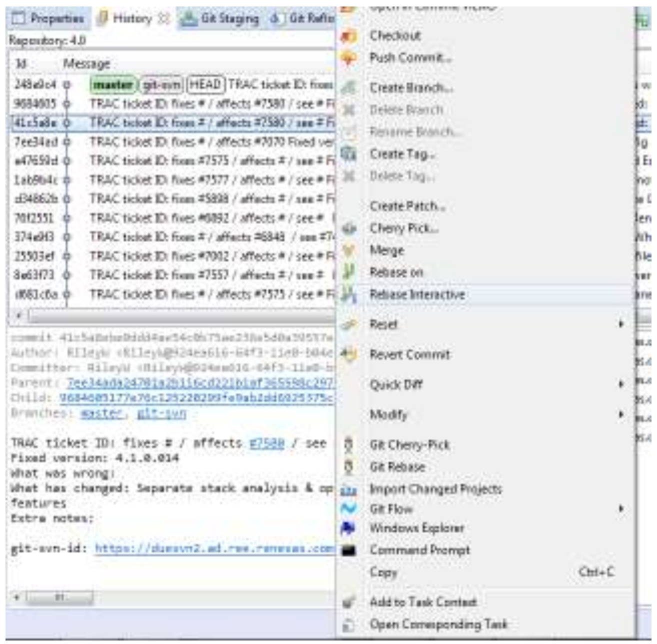
Remove a bad commit from a series of commits

Switch to the local branch containing your change.

Go to history

Right click on the commit before the “bad” one and select “Rebase interactive”

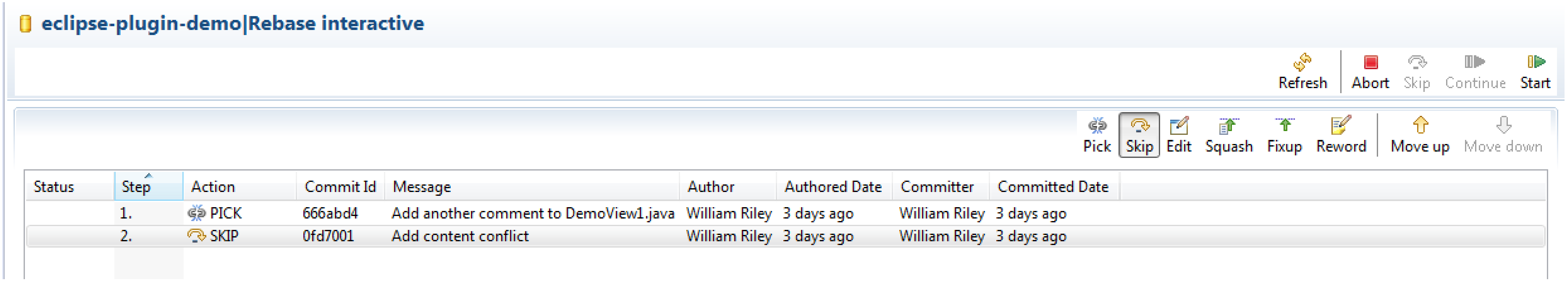
Follow rebase instructions



```
248a9c4  master git-svn HEAD TRAC ticket ID: fixes
9684605  TRAC ticket ID: fixes # / affects #7580 / see # Fi
41c5a8e  TRAC ticket ID: fixes # / affects #7580 / see # Fi
7ee34ed  TRAC ticket ID: fixes # / affects #7070 Fixed ver
a47659d  TRAC ticket ID: fixes #7575 / affects # / see # Fi
1ab9b4c  TRAC ticket ID: fixes #7577 / affects # / see # Fi
d34862b  TRAC ticket ID: fixes #5898 / affects # / see # Fi
70f2551  TRAC ticket ID: fixes #6892 / affects # / see #
374a9f3  TRAC ticket ID: fixes # / affects #6848 / see #7
25503ef  TRAC ticket ID: fixes #7002 / affects # / see # Fi
8a63f73  TRAC ticket ID: fixes #7557 / affects # / see #
d681c6a  TRAC ticket ID: fixes # / affects #7575 / see # Fi

commit 41c5a8ebe0dd4ae54c0b75ae238a5d0a39557e
Author: RileyW <RileyW@924ea616-64f3-11e0-b04c
Committer: RileyW <RileyW@924ea616-64f3-11e0-b
Parent: 7ee34ada24701a2b116cd221b1af365598c297
Child: 9684605177e76c125228299fe0ab2dd6025575c
Branches: master, git-svn

TRAC ticket ID: fixes # / affects #7580 / see
Fixed version: 4.1.0.014
What was wrong:
What has changed: Separate stack analysis & op
features
Extra notes:

git-svn-id: https://duesvn2.ad.ree.renesas.com
```

Remove a bad commit from a series of commits

- In the “Rebase Interactive” view
- Click the commit you wish to skip and click “Skip”
- Click Start

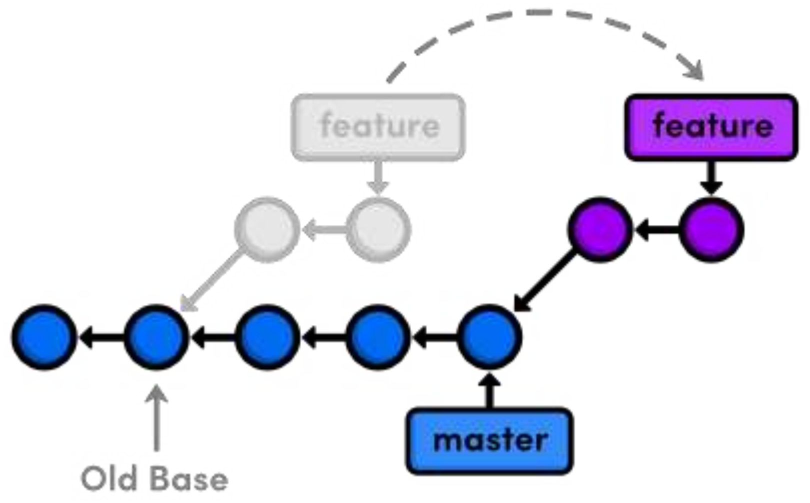


Status	Step	Action	Commit Id	Message	Author	Authored Date	Committer	Committed Date
	1.	PICK	666abd4	Add another comment to DemoView1.java	William Riley	3 days ago	William Riley	3 days ago
	2.	SKIP	0fd7001	Add content conflict	William Riley	3 days ago	William Riley	3 days ago

Rebasing

Rebasing

- Rebasing is the process of moving a branch to a new base commit. The general process can be visualized as the following:



Rebasing

- This is re-writing the history of the repository
- Very powerful
  - Allows commits to be removed completely from the history
- You should only do this to changes not pushed to a remote repository (except when pushed to Gerrit for review)

Interactive Rebasing

- Not going to be covered here
- A standard rebase changes the parent commit and then recreates all the commits
- An interactive rebase allows you to control the recreation process
- This allows you to completely re-write a set of commits
  - Change the order
  - Remove commits
  - Combining
  - Changing message
  - Changing content
- If you want to learn more see: http://www.vogella.com/tutorials/EclipseGit/article.html#interactiverebase_interactiverebaseview