How can you contribute to Eclipse?

Eclipse Con France 2017

21 June 2017
Table des matières

I - How can you contribute to Eclipse ?

5
How can you contribute to Eclipse?

Welcome / Agenda

- general information about contribution
- introduction to requirements (how to install, ...) and tooling
- some words about gerrit
- explanation of the recipe document
- live bug fixing

Who?

Olivier Prouvost:
- OPCoach (www.opcoach.com)
- Eclipse expert
- Provides training and consulting
- France / Europe ... or even further
- Committer on e4 tools and platform UI

Who can contribute?

Everybody!
You only need to open your account on eclipse.org.

Are you a simple User?

- fill bugs on bugzilla: http://bugs.eclipse.org
- ask or answer questions in forums: http://www.eclipse.org/forum
- listen to the mailing lists: http://www.eclipse.org/mail
- connect the automatic reporter
  - it is possible to report anonymous information

1 - http://bugs.eclipse.org
2 - http://www.eclipse.org/forum
3 - http://www.eclipse.org/mail

Contribute to Eclipse, Eclipse Con France 2017
Do you have a development skill? ... but... not yet a committer?

- comment bugs on bugzilla
- start to fix your first bugs!
  - you don’t need to be a committer
  - but be sure to have the git skill!
- what will be your benefits:
  - enhance your development skill
  - improve your visibility as a developer
- attend hackathons
  - to meet the committer and fix your bugs
- propose a CQ (Contribution Questionnaire)
  - for code hosted on github
  - something great to add to Eclipse
  - propose the CQ here: https://dev.eclipse.org/ipzilla/
- become a committer!

4 - https://dev.eclipse.org/ipzilla/
Are you a committer?

➢ open, triage, close bugs
➢ review and accept patches
➢ interact more with the community

What should you do to start your developer contributions?

Initial pre requisites:
➢ Sign the CLA
➢ Open an account on gerrit
➢ Install your Eclipse environment
➢ Read also this wiki page:
  ➢ https://wiki.eclipse.org/Platform_UI/How_to_Contribute

For each contribution
➢ Select a bug to fix
➢ Clone the repository
➢ Fix and test your update
➢ Push your code fix

Contributor License Agreement (CLA)

➢ The CLA:
  ➢ you have authored 100% of the content
  ➢ public record of the contribution
  ➢ you have the rights to contribute (from your company)
  ➢ https://www.eclipse.org/legal/CLA.php
➢ you should have an account on www.eclipse.org -> on the top of the page
➢ be sure to have signed the CLA. To sign it go on this page: https://projects.eclipse.org and click on CLA on the top of the page

Open your gerrit account

Go to: https://gerrit.eclipse.org

Gerrit is used to:
➢ review all patches
➢ manage the global history of modification (on top of git)

Install your Eclipse environment

To fix bug you must:
➢ install the latest Eclipse IDE version for committers (Oxygen)
➢ have a git command in a shell (nice to have) from: https://git-scm.com/download
➢ download the latest target platform from: http://download.eclipse.org/eclipse/downloads
  ➢ for instance the 'I20170612-0950'
  ➢ set it as target platform

5 - https://wiki.eclipse.org/Platform_UI/How_to_Contribute
6 - https://www.eclipse.org/legal/CLA.php
7 - www.eclipse.org
8 - https://projects.eclipse.org
9 - https://gerrit.eclipse.org
10 - https://git-scm.com/downloads
11 - http://download.eclipse.org/eclipse/downloads
- set also as API base line

- you can add also other tools that concern your bugs (cdt, ...)

**Code clone Prerequisites**

Of course you must clone the project git repository:

- go to gerrit
- log in on gerrit web site
- click on the project tab
- search for the project you want to work with
- at the top you will find the ssh git clone commands

12 - http://gerrit.eclipse.org
Clone from gerrit

**Code installation prerequisites**

- Install the repository in the git repository view:
  - add local repository
  - or copy the git repository URL in the view
- Import the projects in the workspace
  - projects are organized with features/bundles...
Find 'where' in the code

You can find 'where' to fix a bug with:

- Plug-in spy (Alt Shift F1)
- Menu Spy (Alt Shift F2)
- Model spy (Alt Shift F9)
- Stack trace in .metadata/.log file
- Bugzilla (helpwanted, bugday...) or by adding a comment
**Doing your fix**
- update the code
- update the header in file (copyright, author), optional
- test your fix!
- Launch another Eclipse with your fix
- Write a test (cf: https://wiki.eclipse.org/Platform_UI/How_to_Contribute#Unit_Testing)

**Gerrit: the code review process**
- Gerrit is used to review the code

13 - https://wiki.eclipse.org/Platform_UI/How_to_Contribute#Unit_Testing
➢ It is an intermediate location where the code is pushed before the merge in the master branch
➢ For a new commit, a review is created and reviewer can comment \((-2,-1,0,+1,+2)\)
➢ The first review is an automatic review done by hudson!
➢ When all reviewers are Ok the code is merged in the master branch
➢ Information on gerrit: [https://wiki.eclipse.org/Gerrit](https://wiki.eclipse.org/Gerrit)
➢ A sample list of reviews (on e4.tools):

<table>
<thead>
<tr>
<th>Subject</th>
<th>Status</th>
<th>Owner</th>
<th>Project</th>
<th>Branch</th>
<th>Updated</th>
<th>Size</th>
<th>CPF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oxygen Version increase for changed org.eclipse.e4tools.* files</td>
<td>Merged</td>
<td>Oliver Croquette</td>
<td>org.eclipse.e4tools</td>
<td>master</td>
<td>Jun 1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>Bag 513535 - Use consistent UI settings for search input in spec</td>
<td>Merged</td>
<td>Lars Vogel</td>
<td>org.eclipse.e4tools</td>
<td>master</td>
<td>Jun 1</td>
<td>-2</td>
<td>-2</td>
</tr>
<tr>
<td>Bag 513515 - The biases menu is not displayed reliably</td>
<td>Merged</td>
<td>Lars Vogel</td>
<td>org.eclipse.e4tools</td>
<td>master</td>
<td>Jun 1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>Bag 513494 - Many of the plugins need a version increase</td>
<td>Merged</td>
<td>Lars Vogel</td>
<td>org.eclipse.e4tools</td>
<td>master</td>
<td>Jun 1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>Bag 514517 - Does obsolete content of the old-e4 tools wizard exist</td>
<td>Merged</td>
<td>Lars Vogel</td>
<td>org.eclipse.e4tools</td>
<td>master</td>
<td>Jun 1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>Bag 515635 - Do not set the CI to the user's account in the selected error</td>
<td>Merged</td>
<td>Lars Vogel</td>
<td>org.eclipse.e4tools</td>
<td>master</td>
<td>Jun 1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>Bag 515635 - Do not set the CI to the user's account in the selected error</td>
<td>Merged</td>
<td>Lars Vogel</td>
<td>org.eclipse.e4tools</td>
<td>master</td>
<td>Jun 1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>Bag 515635 - Do not set the CI to the user's account in the selected error</td>
<td>Merged</td>
<td>Lars Vogel</td>
<td>org.eclipse.e4tools</td>
<td>master</td>
<td>Jun 1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>Bag 515635 - Do not set the CI to the user's account in the selected error</td>
<td>Merged</td>
<td>Lars Vogel</td>
<td>org.eclipse.e4tools</td>
<td>master</td>
<td>Jun 1</td>
<td>+1</td>
<td>+1</td>
</tr>
</tbody>
</table>

**Review contents**

➢ A review contains comments and patches
➢ The ID of the review is used to get the code in your repository

---

14 - [https://wiki.eclipse.org/Gerrit](https://wiki.eclipse.org/Gerrit)
Fetch from gerrit

➢ It is possible to import any review in your eclipse
➢ Use the Team->Remote->Fetch From Gerrit command
Fixing bugs can put the developers in tricky situations

To help you to face these problems I prepared a recipe notebook

It contains the usual use cases:

You can download it here:


There are also other resources:

french: http://alain-bernard.developpez.com/tutoriels/eclipse/contribuer

eclipse: https://wiki.eclipse.org/Platform_UI/How_to_Contribute
Evaluate the Sessions

Sign in and vote at eclipsecon.org

-1  0  +1