Eclipse Committer Bootcamp

Wayne Beaton - Open Source Projects

Janet Campbell - Intellectual Property

Denis Roy - Information Technology
Eclipse Committer Bootcamp

Part I
Exploiting the
Eclipse Development Process
for Fun and Profit

http://eclipse.org/projects/dev_process
Agenda

• Open source rules of engagement
• Starting a Project
• Projects, Code, and Resources
• Who's Who
• Project Management Infrastructure
Open Source Rules of Engagement

- Transparency
- Openness
- Meritocracy
Transparent: Invite Participation

• Project discussions, plans, minutes are open, public, and easily accessible
  - Use “dev” list for project-related discussion
  - Capture all work in Bugzilla records

• Private team meetings, issue trackers are OK
  - Capture ideas, issues and discussions in public
Open: Accept Participation

- The same opportunity to all
- Everyone participates with the same rules
- There are no rules to exclude any potential contributors
  - Including direct competitors in the marketplace
Meritocracy: Earn your Way in

• The more you contribute the more responsibility you will earn
• Leadership roles in Eclipse are also merit-based and earned by peer acclaim
Three Communities

• Users
  - Users are, well... users

• Adopters
  - Individuals, groups, organizations
  - Build products, extensions, based on your project

• Developers
  - Contributors, committers
Community takes time

- Building and maintaining a community is hard work, takes time
- Often difficult to sell to management/bean counters
- A large, successful community pays off for the project
Project Hierarchy

- Top Level Project
  - Project
    - Project (Subproject)
  - Project
  - Top Level Project
    - Project
      - Project (Subproject)
    - Project (Subproject)
      - Project (Subproject)
Projects, Committers, and Resources

Resources and Committers

<table>
<thead>
<tr>
<th>Resources</th>
<th>Committers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Website</td>
<td>Bugzilla</td>
</tr>
<tr>
<td>Git</td>
<td>IPZilla</td>
</tr>
<tr>
<td>Downloads</td>
<td>Mailing Lists</td>
</tr>
<tr>
<td>HIPP</td>
<td>Forums</td>
</tr>
<tr>
<td>Metadata</td>
<td></td>
</tr>
</tbody>
</table>

Top-Level Project

Project

Project (Subproject)

Resources and Committers

Project (Subproject)

Resources and Committers
Leadership Chain

1. Board of Directors
2. EMO(ED)
3. Project Management Committee
4. Project Leads
Project Members

• Project Management Committee (PMC)
  - Top-level oversight, IP process, various approvals

• Project Lead(s)
  - Leadership of the project

• Committers
  - Eclipse IP Due Diligence Process
  - Eclipse Development Process
Eclipse Management Organization (EMO)

- Eclipse Foundation Staff
- Architecture Council
- Planning Council
- EMO (ED): Executive Director
- Email: emo@eclipse.org
Architecture Council

• Stewards of the Eclipse Development Process
• Architectural oversight
• Best practices
• Mentors for new Eclipse projects
  - Use your mentors!
Planning Council

- Simultaneous Release
- Cross-project planning
- Architectural issues
- User interface conflicts
- Other coordination and integration issues
Project Creation Process

- Project Proposal
  - Community Review
  - Trademark Review
- Creation Review
  - Provisioning
  - Initial Contribution
- Plan, Develop, Review, Release
Project Proposal

• Create a proposal
  - https://projects.eclipse.org/create/project-proposal

• Feedback cycle

• Approval from EMO(ED)
  - Scope, description, interested parties, committers
Community Review

- Minimum of two weeks of community review
- Identify two mentors from the Architecture Council
- Respond to questions from community
- Identify additional committers
- Refine the proposal
Creation Review

• Secure project name trademark
• One week final community review
• Respond to questions from community
Provisioning

• Submit provisioning request
• IP Team handles committer paperwork
• Webmaster team provisions project resources
Initial contribution

- Initial contribution reviewed by IP team
- Submit in the form a ZIP file
- Commit nothing to VCS until approved
Some Sharing

• A parent project may share:
  ‧ Builds, Downloads
  ‧ Website, mailing lists, and forums

• May not share:
  ‧ Committers
  ‧ Repositories, Bugzilla
Project Metadata

• Project id
  - e.g. technology.egit, soa.winery, eclipse.jdt.ui
• Description, scope, logo, technology type
• Releases
• Relationships to other projects
• Build technologies
Eclipse Git Team Provider

EGit is an Eclipse Team provider for Git. Git is a distributed SCM, which means every developer has a full copy of all history of every revision of the code, making queries against the history very fast and versatile. The EGit project is implementing Eclipse tooling for the JGit Java implementation of Git.

Download Eclipse Git Team Provider

Marketplace:
EGit in Eclipse Marketplace

Update Sites:
EGit p2 repository - latest release: http://download.eclipse.org/egit/updates
EGit p2 repository - nightly build: http://download.eclipse.org/egit/updates-nightly
EGit p2 repository - nightly maintenance build: http://download.eclipse.org/egit/updates-stable-nightly

Forums:
eclipse.egit [vmp]
Questions and technical discussions about how to use the Git team provider (EGit)
PMI: Edit Project Metadata

Home » Technology Project » Eclipse Git Team Provider

Project Name *
Eclipse Git Team Provider

› THE BASICS

› CATEGORIZATION, RELATIONSHIPS, AND TAGS

› SOURCE CODE

› BUILD

› DOWNLOADS

› LINKS

› COMMUNICATION
PMI: Categorization

**CATEGORIZATION, RELATIONSHIPS, AND TAGS**

**Technology Types**
- [ ] Language
- [ ] Machine-to-Machine
- [ ] Modeling
- [ ] OSGi
- [ ] Runtime
- [ ] Testing
- [x] Tools

Select the types of technology *produced* by the project.

**RELATED PROJECTS**

Specify any projects that are related to this project. What "related" means is really up to you. The values you specify here will rendered on the project page as hyperlinks.

[Add another item]
PMI: Source Code

- Contribution Message
- Bugzilla
- Source Repositories
PMI: Build

- Description
- Build Technologies
- Documentation
- Links
PMI: Downloads

- “Big Button” URL
- Message
- Marketplace
- Update sites
- Downloads
Links and Stuff

• Important Mailing Lists
  - cross-project-issues-dev
  - PMC, Project Dev

• Development Resources

• The Eclipse Development Process
  - http://eclipse.org/projects/dev_process

• The Project Management Infrastructure
  - http://wiki.eclipse.org/Project_Management_Infrastructure

• Starting a Project
The Eclipse Intellectual Property Process and You

Part II

Janet Campbell
Director
Intellectual Property Management
AGENDA

Eclipse Intellectual Property Overview
Meet the Eclipse IP Team
Benefits
Origin of Eclipse Code
Necessary Rights
Due Diligence Process Poster
IPzilla
INTELLECTUAL PROPERTY

… a legal concept which refers to creations of the mind for which exclusive rights are recognized!
Eclipse Intellectual Property Management

Eclipse IP Policy …

Designed to ensure we maintain the cleanliness of the Eclipse code base…

• Necessary Rights
• Non Restrictive Licenses
Meet the Team

Janet Campbell

Sharon Corbett
Coming Soon😊

To reach the team email emo-ip-team@eclipse.org
Benefits of the Eclipse IP Management

- Risk Reduction
- Collaboration
- Adoption
Origin of Eclipse Code

Code originates from one of the following three sources:
Contributions from Eclipse Committers
Contributions from Contributors
Contributions from third party sources (e.g., another open source project)
Securing Necessary Rights

Committers:
  • Individual Committer Agreement
  • Employer Consent Form (if individual is employed)
  • Member Committer Agreement

Contributors:
  • Eclipse Contribution License Agreement (CLA)

Third Party Sources (another open source project):
  • Project Contribution Mechanism in place?
Due Diligence Process
Continued

In Part V, we will dive into the Eclipse Due Diligence Services in greater detail!
Part III: Provisioning & Server Resources
http://wiki.eclipse.org/IT_Infrastructure_Doc
Agenda

The team
Server infrastructure overview
You Eclipse Foundation account, committer ID
Project provisioning process
Committing your initial contribution
Interacting with users and other developers
Asking for help
Your Webmaster Team

Matt Ward – Server Samurai

??? – Build Guru / Git Ninja

Denis Roy – Hey, that's me!

Web Developers: Chris Guindon
& Edouard Poitras

webmaster@eclipse.org
Server Infrastructure

3 Cabinets in Ottawa, Canada
60 kW redundant AC power
1 Gbps backends
1 Gbps BGP-4 bandwidth (rate limited)
45 TB/month
45M web pages/month (www & wiki)
Download servers: 9M files/day (14M hits)
~60 download mirrors worldwide
99.995% service availability
Eclipse Foundation Forges

Eclipse Is...

OPEN SOURCE LOCATION AWARE TECHNOLOGY FOR THE ENTERPRISE

Grow you Community, Grow your Business
High performance event processing
Big Data Analytics
Mobile Applications
Open to All, Mature Governance with High Transparency

PolarSys
Open Source Tools for Embedded Systems

PolarSys Working Group
- Open Innovation to create better methods and tools
- Software tools for critical systems
- Interoperability based on Open Standards
- Fostering of exchanges between academics and industrial partners
- Quality and maturity assessment
- Very Long Term Support for more than 10 years

Component Highlight
TOPCASED migrates to PolarSys

Events
PolarSys networking event in Montréal
Accounts

Committer ID vs. Email ID
Committer ID for SSH, Gerrit (even https)
Email for everything else
This is Open Source -- Email addresses are shown!
Accounts are shared across all our forges

This reverts commit 7c5b2a07e173ab5551549c30571740504b312cbee.

```diff
--- a/bundles/org.eclipse.ui.workbench/Eclipse UI/org.eclipse.ui/internal/WorkbenchPage.java
+++ b/bundles/org.eclipse.ui.workbench/Eclipse UI/org.eclipse.ui/internal/WorkbenchPage.java
@@ -192,14 +192,14 @@ public class WorkbenchPage implements IWorkbenchPage {
     @Override
     public void partActivated(MPart part) {
         updateActivations(part);
-        firePartActivated(part);}
+        firePartActivated(part);}
```

http://git.eclipse.org
Gerrit Code Review

Any contributor can push to Gerrit repository
Review/vote before merging with master
Committer votes
Hudson "votes"

http://git.eclipse.org/r
Contributions and Community

Contributions come in through Bugzilla or Gerrit

CLA (Contributor License Agreement)
CLA

http://projects.eclipse.org
Issue Tracker: Bugzilla

Bug 401288 - Require possibility to specify workspace (data) directory of tests to run. (edit)

Reported: 2013-02-20 05:16 EST by Johann Draschwendtner

Modified: 2013-05-03 08:26 EDT (History)

CC List: Add me to CC list
4 users (edit)

See Also: (add)

Attachments
- proposedfix (1.34 KB, patch) no flags

Additional Comments:

- Status: NEW
- Mark as Duplicate

Pedro Laros - CLA 2012-07-27 13:00:26 EDT

Build Identifier: 2.0.4

SWTBotTable.columnCount returns 0 when there are no columns, or when a TableColumn has not been explicitly created. This causes the selection method to return a TableRow item with no text even though the selection is valid in the table widget.

Reproducible: Always
## User Community: Forums

<table>
<thead>
<tr>
<th>Topic</th>
<th>Replies</th>
<th>Views</th>
<th>Last message</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Announcement: Android Development Info ** READ FIRST ****&lt;br&gt;ADT (Android Development Tools) users - your answers are here!</strong>&lt;br&gt;By: Eric Rizzo on Wed, 27 July 2011</td>
<td>0</td>
<td>150272</td>
<td>Wed, 27 July 2011 17:54&lt;br&gt;By: Eric Rizzo ●</td>
</tr>
<tr>
<td><strong>Announcement: Eclipse won't start? READ THIS FIRST</strong>&lt;br&gt;By: Eric Rizzo on Fri, 15 October 2010</td>
<td>0</td>
<td>59779</td>
<td>Fri, 15 October 2010 14:03&lt;br&gt;By: Eric Rizzo ●</td>
</tr>
<tr>
<td><strong>Announcement: New forum members, READ THIS FIRST,</strong>&lt;br&gt;By: Russell Bateman on Sun, 10 October 2010</td>
<td>0</td>
<td>37733</td>
<td>Sun, 10 October 2010 01:24&lt;br&gt;By: Russell Bateman ●</td>
</tr>
<tr>
<td><strong>Announcement: Linux + Eclipse users, read this!</strong>&lt;br&gt;Avoid many problems when running Eclipse on Linux&lt;br&gt;By: Eric Rizzo on Fri, 22 January 2010</td>
<td>0</td>
<td>29561</td>
<td>Fri, 22 January 2010 15:07&lt;br&gt;By: Eric Rizzo ●</td>
</tr>
<tr>
<td><strong>Announcement: C/C++ Eclipse User Info</strong>&lt;br&gt;There is a forum for CDT -- this is not it.&lt;br&gt;By: Eric Rizzo on Fri, 22 January 2010</td>
<td>0</td>
<td>61015</td>
<td>Fri, 22 January 2010 14:37&lt;br&gt;By: Eric Rizzo ●</td>
</tr>
<tr>
<td><strong>Locking is not possible in the directory</strong>&lt;br&gt;Problem with locking on startup&lt;br&gt;By: Margaret doll on Fri, 29 March 2013</td>
<td>2</td>
<td>11923</td>
<td>Thu, 23 October 2014 18:58&lt;br&gt;By: Ab B ●</td>
</tr>
<tr>
<td><strong>Question to Unreachable Code</strong>&lt;br&gt;Code is not reachable, but there's no warning&lt;br&gt;By: Antonio Bloise on Thu, 23 October 2014</td>
<td>0</td>
<td>205</td>
<td>Thu, 23 October 2014 13:46&lt;br&gt;By: Antonio Bloise ●</td>
</tr>
<tr>
<td><strong>Stand Alone Java App Servlet development...</strong>&lt;br&gt;By: Jim Crowell on Wed, 01 October 2014</td>
<td>3</td>
<td>579</td>
<td>Thu, 23 October 2014 12:59&lt;br&gt;By: Eric Rizzo ●</td>
</tr>
</tbody>
</table>
Developer Community: Mailing Lists

Mailing list: jubula-dev
Jubula platform and tools development

About jubula-dev
Jubula platform and tools development

Using jubula-dev
To post a message to all the list members, send email to jubula-dev@eclipse.org. You must be subscribed to the list before you can post. To access a web archive of this list, visit the jubula-dev Archives or subscribe to this list's RSS feed.

Subscribing to jubula-dev
All contributions you make to our web site are governed by our Terms Of Use. Your interactions with the Eclipse Foundation web properties and any information you may provide us about yourself are governed by our Privacy Policy.

Subscribe to jubula-dev by filling out the following form. You will be sent email requesting confirmation, to prevent others from gratuitously subscribing you. This is a hidden list, which means that the list of members is available only to the list administrator.

<table>
<thead>
<tr>
<th>Your email address:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Your name (optional):</td>
</tr>
</tbody>
</table>
Project Provisioning process

Project space: Git, www.eclipse.org, Bugzilla, Mailing Lists, Forum

Committing IP-approved Initial Contribution to git.eclipse.org

Culling history on Github-hosted repos
Eclipse Java development tools (JDT)

About JDT

The JDT project provides the tool plug-ins that implement a Java IDE supporting the development of any Java application, including Eclipse plug-ins. It adds a Java project nature and Java perspective to the Eclipse Workbench as well as a number of views, editors, wizards, builders, and code merging and refactoring tools. The JDT project allows Eclipse to be a development environment for itself.

More details...

JDT Components

The JDT project is broken down into components. Each component operates like a project unto its own, with its own files, perspectives, and configuration options.
Interacting With Others

Dev lists for committers (typically)
Forums for user discussions (again, typically)
Eclipse.org-committers mailing list
Cross-project-issues-dev mailing list
Bugzilla – Eclipse Foundation > Community
webmaster@eclipse.org (Servers & Infra)
emo@eclipse.org (Project, community & process)
Eclipse Committer Bootcamp

PART IV

The Eclipse Development process

http://eclipse.org/projects/dev_process
Agenda

- Community development
- Elections
- Releases, plans, and reviews
- IP Logs
Writing code is fun, but...

• Community is important!
• Have project-specific diversity goals
  - Building diversity takes work
• Actively court contributors
  – Use Gerrit for easy contribution workflow
  – Make it easy for contributors – spell it out clearly
• Be responsive when they do come
  - “Kill with kindness”
Pragmatically Speaking...

• Keep project information up-to-date
  - Project and release metadata, website, downloads
• Project code must be buildable
• Have a contribution guide
  - “Getting started”
  - CONTRIBUTING file in project repositories
  - https://bugs.eclipse.org/397644
Contribution Guide

- Git, Gerrit, GitHub, ...
- Issue tracking (Bugzilla)
  - “Help wanted” issues
- Project plan
- How to build
- How to engage (mailing lists, forums, Bugzilla)

Contributing to Eclipse Platform

Thanks for your interest in this project.

Project description:

The Platform defines the set of frameworks and common services that collectively make up infrastructure required to support:

- https://projects.eclipse.org/projects/eclipse.platform
Outreach

- Present at conferences
- Social media: Blog, tweet, ...
- Author papers, articles, ...
- You know your community best
  - Where do they hang out?
- Help potential contributors find you
  - Serving one community can build another
  - e.g. A happy user community builds the adopter community; a large adopter community drives contribution
Spend Time With the Community
Meritocracy

• Nominees need to prove themselves
• How much merit is enough?
  - Project-specific (work with your PMC)
  - Tends to be qualitative, not quantitative
• Nomination criteria:
  - Source code contributions
  - Forum activity
  - Subject matter expert
  - ...

Election Workflow

1. Ongoing
   - Committer vote
   - One week or all committers have voted

2. Complete
   - Committers vote +1
   - Not enough votes
   - PMC Approval
     - Timed Out
     - Approved
     - Vetoed
     - Rejected
   - Commiter votes -1
Releases*

* "Releases" doesn't mean nightly or milestone builds – these builds don't
Project Plans

- **Description**
  - Paragraph, no-bullets preferred
- **Milestones**
  - Name, date, description
- **Themes**
  - Bugzilla URLs
- **Deliverables, Compatibility, Target Environments, Internationalization**

**Description:**
The BPMN2 Modeler is a graphical modeling tool which allows creation and editing of BPMN 2.0 compliant diagrams. The tool is built on Eclipse Graphiti and uses the MDT BPMN2 project as the underlying model. This release represents the first stable version of the editor. Version 1.0, while still not a final, polished product, is very stable and offers a very complete API that achieves the goals set for this release. The project leadership would like to thank the university researchers and community users who helped define and refine the editor API you now use. (You are and for making BPMN2 Modeler a better product.)

**Deliverables:**
- GitMnt BPMN2 model
- BPMN extension plug-in
- Code samples and tutorials

**Compatibility:**
This, and all releases going forward, will only support Graphiti version 0.10.x and higher. The Graphiti project releases a new version with breaking API changes. BPMN2 Modeler will be updated to support those new versions of Graphiti.

- A new extension point has been added to allow an extension plug-in to provide its own Graphiti Tool Behavior. See [Bug 115346] for details.
- The class hierarchy for Custom Tasks has been restructured to allow extension plug-ins to define custom extensions for BPMN 2.0 connection elements as well as shapes. See [Bug 115732] for details.

**Internationalization:**
String translation for all UI messages will be addressed in the next release scheduled for end of Q4, 2013.

**Target Environments:**
This release requires Java 6 and is targeted for Kepler. Testing has been done on the following hardware/OS platforms:
- MS-Windows 7
- Fedora/RHEL 13
- Mac OS X 10.8 Mountain Lion

<table>
<thead>
<tr>
<th>Name</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML</td>
<td>2012/10/15</td>
<td>Initial Contribution</td>
</tr>
<tr>
<td>M2</td>
<td>2012/10/15</td>
<td>Milestone Build</td>
</tr>
<tr>
<td>PC1</td>
<td>2012/10/30</td>
<td>Release Candidate for 0.1.0</td>
</tr>
<tr>
<td>0.1</td>
<td>2012/10/03</td>
<td>Code Stabilization Release</td>
</tr>
<tr>
<td>0.2</td>
<td>2012/12/12</td>
<td>API Stabilization Release</td>
</tr>
<tr>
<td>0.21</td>
<td>2013/01/12</td>
<td>Bug FixRelease</td>
</tr>
<tr>
<td>0.22</td>
<td>2013/02/07</td>
<td>Bug FixRelease</td>
</tr>
<tr>
<td>0.23</td>
<td>2013/03/15</td>
<td>Bug Fixes and New Example</td>
</tr>
<tr>
<td>0.24</td>
<td>2013/04/09</td>
<td>Bug Fixes and New Features</td>
</tr>
<tr>
<td>0.25</td>
<td>2013/04/25</td>
<td>Bug Fixes and New Features</td>
</tr>
<tr>
<td>0.26</td>
<td>2013/06/07</td>
<td>Bug Fixes and New Features</td>
</tr>
<tr>
<td>0.27</td>
<td>2013/08/30</td>
<td>Bug Fixes and New Features</td>
</tr>
<tr>
<td>1.0</td>
<td>2013/11/15</td>
<td>1.1 Milestone 1</td>
</tr>
<tr>
<td>1.0</td>
<td>2013/12/30</td>
<td>1.1 Release and Graduation</td>
</tr>
<tr>
<td>1.0</td>
<td>2013/12/01</td>
<td>String Externalization, Internationalization and Documentation</td>
</tr>
</tbody>
</table>

**Themes:**
Graduation
This version marks the first stable release of BPMN2 Modeler. See the Architecture/Issues section for a discussion of the changes made in this release.
Defining a New Release*

* “Releases” doesn't mean nightly or milestone builds – these builds don’t
Release Review

- Assemble Review Doc
- Assemble IP Log

-2 weeks
 PMC Review Documentation
 IP Team Review IP Log

-1 week
 Start Release Review

End Release Review

Publish Release
Release Review

• Major/minor releases
  - Release review
  - IP Log approval
  - Plan to spend time planning/documenting release

• Service/Bugfix-only releases
  - No review
  - No IP Log approval
Release Naming

• `<major>`.`<minor>`.`<service>`
  - e.g. 0.3, 1.2.4, ...

• Release
  - e.g. 0.7

• Milestones
  - Use the expected release name with M/RC
  - e.g. 0.7M1, 0.7M2, 0.7M3, 0.7RC1, 0.7RC2
  - Not for general public consumption
Incubation

• Releases conventionally use pre-1.0 names
• Incubation branding
  - Incubation logo on their project home and primary download pages
  - Downloads include the word "incubation" in the filename
    • Not required for JAR files
  - Bundle and feature names include the word "incubation"
    • Not required for "Bundle-SymbolicName"s
• Incubation ends with a graduation review
  - Generally combined with a release review
• Mature projects may request a permanent incubator subproject
Other Reviews

• Graduation
  - Generally combined with a release review
  - Demonstrate committer familiarity with EDP/IP policy
  - API stability, quality code

• Termination
  - Lack of development resources, will, interest, ...
  - Done?
IP Logs

• The IP Log Generator
• IP Log review process
The IP Log Generator

- Git/SVN
- Committer Activity (Dash)
- Repositories (PMI)
- Contributions (Dash)
- Licenses (Foundation DB)

IP Log Generator

- Contributions (Bugzilla)

IP Log
commit d6cf52411377a039fc2906378711091a26e932cb
Author: Some Body <somebody@somewhere.com>
Date:   Wed May 29 16:17:36 2013 +0200
    Hide unwanted common navigator action bar items

This change hides unwanted 'Link with Editor' and 'Customize View...' items from the local toolbar and the view menu.

See bug 409722 for restoring the feature of activating the editor when changing the selection using 'Link with Selection'.

Bug: 350686
Change-Id: Ia2bd5091303d1b0a738157effc24e4dac5a7d0c7
Also-by: Some Bodyelse <somebodyelse@nowhere.com>
Signed-off-by: Some Body <somebody@somewhere.com>
The Download Scanner

- Linked from project page's "Action" menu
- Validates third-party library use in project download directories
- Limited to Java/OSGi-based files
- Should be considered a guide
  - Committers are responsible for following the Eclipse IP policy and process
- Don't count on this tool to get it exactly right

http://eclipse.org/projects/tools/downloads.php?id=<projectId>
The Eclipse Intellectual Property Process and You

Janet Campbell
Director
Intellectual Property Management
Agenda

Due Diligence Overview
Review Submissions (aka CQs)
Committer’s Role
Reuse Possibilities?
Parallel IP
The Review
Best Practices
Legal Resources
Getting in Touch
Eclipse Due Diligence

Requests for Due Diligence Reviews are initiated by Committers

Committers adhere to the Eclipse Due Diligence Process

Requests are submitted via a Contribution Questionnaire (fondly known as a CQ)
Contribution Questionnaires (aka Submissions)

Project Code
Hosted/Maintained @ Eclipse
EPL (Typical) License
Dual Licensed Scenarios Possible

Non Project Code (Third Party)
Not Hosted/Maintained @ Eclipse
Various Licenses
Committer Knowledge

• Origin/Source
• Provenance Information
• License(s)
• Binary/Source
• Modified/Unmodified
• Attach Source via Zip (Class Files not Required)
• Ensure No Nesting/Narrow Scope where possible
• Project Licensed and Non Project Licensed Code - not reviewed together
Reuse?

Has the package been previously approved for Eclipse distribution?

- Yes – Request Reuse
- No – Request a new CQ
Parallel IP

Initial check performed for provenance/license
Enables projects to get up and running as quickly as possible

Graduated projects are eligible for Mature Parallel only for revisions of previously approved third party libraries
Due Diligence Review

Forensic Analysis
# The Review

<table>
<thead>
<tr>
<th>License</th>
<th>Provenance</th>
<th>Due Diligence</th>
</tr>
</thead>
<tbody>
<tr>
<td>What license applies?</td>
<td>Who wrote the code?</td>
<td>Provenance Concerns?</td>
</tr>
<tr>
<td>Is the license suitable?</td>
<td>How did they agree to the license?</td>
<td>Licensing Concerns?</td>
</tr>
</tbody>
</table>

**Diagram:**
- **License**: What license applies? Is the license suitable?
- **Provenance**: Who wrote the code? How did they agree to the license?
- **Due Diligence**: Provenance Concerns? Licensing Concerns?
Tooling

- Copyright Reference
- Keyword Search
- Borrowed/Copied Code
- IBM Scan Tool/Blackduck Protex
- License References
- Inappropriate Language
- Words with Legal Significance
We’ve Done our Homework

<table>
<thead>
<tr>
<th>Review Complete</th>
<th>Areas of Concern Investigated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Other open source projects contacted if required regarding investigation/resolutions</td>
</tr>
<tr>
<td></td>
<td>If approval is not possible, committer is contacted and advised of Foundation’s concerns and a technical workaround is investigated</td>
</tr>
</tbody>
</table>
IP Best Practices

Follow the DD Process
Understand your code and what the project intends to distribute
Scope/No Nesting
Separate CQs (project licensed/third party)
Third party content must receive green light for checkin or full approval prior to being placed in the Eclipse repository
When in doubt, check in with us (emo-ip-team@eclipse.org)
Eclipse Legal Resources

Legal - www.eclipse.org/legal

Committer Agreements - http://wiki.eclipse.org/Development_Resources/HOWTO/Nominating_and_Electing_a_New_Committer
Getting In Touch

IP Process Questions
emo-ip-team@eclipse.org

License Questions
license@eclipse.org

Committer Legal Agreements
emo-records@eclipse.org
Eclipse Committer Bootcamp

Part VI: Builds & Downloads
http://eclip.se/t
Agenda

Building: Hudson/HIPP, CBI
Signing JAR files
Storage: build artifacts, nightlies, releases
Using mirrors
Download statistics
Cleaning up
Common Build Infrastructure

Hudson CI
Git/Gerrit
Maven/Tycho
Jar signing
Nexus (Maven repository)

http://wiki.eclipse.org/CBI

Make it easy for anyone to build your code!
Committer Tools

Development Tools
- Bugzilla
- Git
- Gerrit code review
- Request Webmaster Support

Committer Tools
- PMI - Project management infrastructure
- IPZilla
- Bugzilla Manager: components, targets, milestones
- Committer mailing list
- Download stats
- Website stats
- Eclipse infras status
- IP address / DNS test tool

Hudson HIPP Control
- Babel: stopped
- Common Build Infrastructure: running
- Dash, Tools for Committers: No HIPP instance found for this project. Request one.
- Request Webmaster support for a HIPP instance
- Hudson and HIPP docs
Hudson & HIPP
Employs Hudson Continuous Integration
Shared Hudson w/ Mac and Windows UI slaves
HIPP: Hudson Instance Per Project
Signing

JAR files: queued and private web service
  Queued for many files, ZIPs
  Web service for on-the-fly signing, jars only
Windows executables via private web service
Mac executables via private web service
Maven signing plugin: http://eclip.se/u
Storing builds

Temporary stores: build artifacts, workspace
Nightly builds not mirrored
Stable & Release: mirrored
Simultaneous Release
Storage is not unlimited!
Maven: repo.eclipse.org
Use Mirrors? Get download stats.
P2 & mirrors

What to add?
The `p2.mirrorsURL` property has the following structure:

```xml
<property name="p2.mirrorsURL"
```

- replace `{repository_path}` with the path where your artifacts.jar sits on download.eclipse.org.

P2 & stats

There are two steps to enable p2 download statistics gathering for your repository:

1) In the artifact repository that you want to track downloads from, add a `p2.statsURI` property specifying the statistics URL (in artifa

```xml
<repository name='Update Site' type='org.eclipse.equinox.p2.artifact.repository.simpleRepository' version='1'>
    <properties size='3'>
        <property name='p2.timestamp' value='1269575706171'/>
        <property name='p2.compressed' value='true'/>
        <property name='p2.statsURI' value='http://your.stats.server/stats'/>
    </properties>
</repository>
```
Download statistics

wiki.eclipse.org/Equinox_p2_download_stats
wiki.eclipse.org/Equinox/p2/p2.mirrorsURL
Ask cross-project-issues-dev for help!
No solution yet for Maven/repo.eclipse.org stats
Stats Tool

Eclipse Download Stats

Download requests are recorded two ways: by using the Find A Mirror script, and by configuring your p2 repository. Direct downloads from mirrors and files linked directly to download.eclipse.org are not included here.

Search Criteria

Partial File Name: eclipse-%-luna-R-%.zip

TIP: For faster results, use a file name that matches the fewest amount of files as possible to satisfy your query. For instance, query using the core file(s) that make up one user download.

Group files
Send results to email:

Run query


Today’s downloads will be added at midnight Eastern Time.

Date

Date From:
Date To:

TP: data-based queries take much longer to run when files with many downloads are included.

View Grouped by country

Results

Query took 21.600 sec (0.001 connect time)?

<table>
<thead>
<tr>
<th>File Group</th>
<th>Code</th>
<th>Country</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cn</td>
<td>China</td>
<td>3695236</td>
</tr>
<tr>
<td></td>
<td>us</td>
<td>United States</td>
<td>670499</td>
</tr>
<tr>
<td></td>
<td>in</td>
<td>India</td>
<td>333830</td>
</tr>
<tr>
<td></td>
<td>kr</td>
<td>Korea, Republic Of</td>
<td>217996</td>
</tr>
<tr>
<td></td>
<td>de</td>
<td>Germany</td>
<td>184972</td>
</tr>
</tbody>
</table>
Cleaning up

Retention policy
Source code and Bugzilla
Hudson build artifacts
Download.eclipse.org
Older builds: archive.eclipse.org
Links

wiki.eclipse.org/IT_Infrastructure_Doc
wiki.eclipse.org/IT_Infrastructure_Doc#Builds
wiki.eclipse.org/Hudson
Maven signing plugin: http://eclipsse/u
https://dev.eclipse.org/site_login/
Eclipse Committer Bootcamp

THANK YOU!

Wayne Beaton – emo@eclipse.org

Janet Campbell – emo-ip-team@eclipse.org

Denis Roy – webmaster@eclipse.org