

IP for Eclipse Committers

Janet Campbell
Director, Intellectual Property
Legal Counsel and Secretary
Eclipse Foundation, Inc.

Outline

1. Due Diligence

- Why do it?
- Who does what?

2. Leveraging Parallel IP

3. Third Party Dependencies

4. Dual Licensing

- Eligibility
- Important Considerations

Due Diligence – Why do it?

- Commercial Entities:



- Developers:



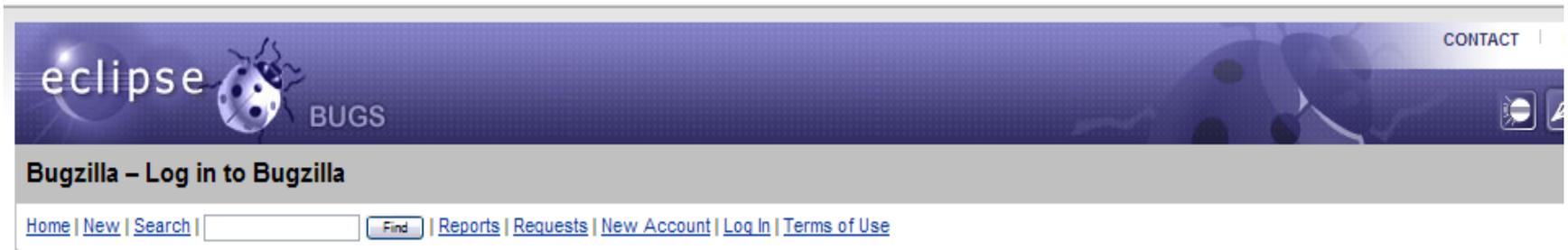
Due Diligence – Who does what?

We all have a part to play:

- All Contributions are reviewed by the **Committer** who wishes to include the code in the Eclipse code base in accordance with the [Eclipse IP Process](#).
- The **PMC** must approve all Contributions for their project.
- The **Project** maintains a [Project Log](#) of all contributions and documents the use of non-EPL code and cryptography in their [About files](#).
- The **Foundation** will also review some Contributions – what is reviewed will depend on the nature of the Contribution (e.g. size, applicable license, use of cryptography) and its origin.

Accepting Contributions

- Contributions must be submitted via a bug tracking system



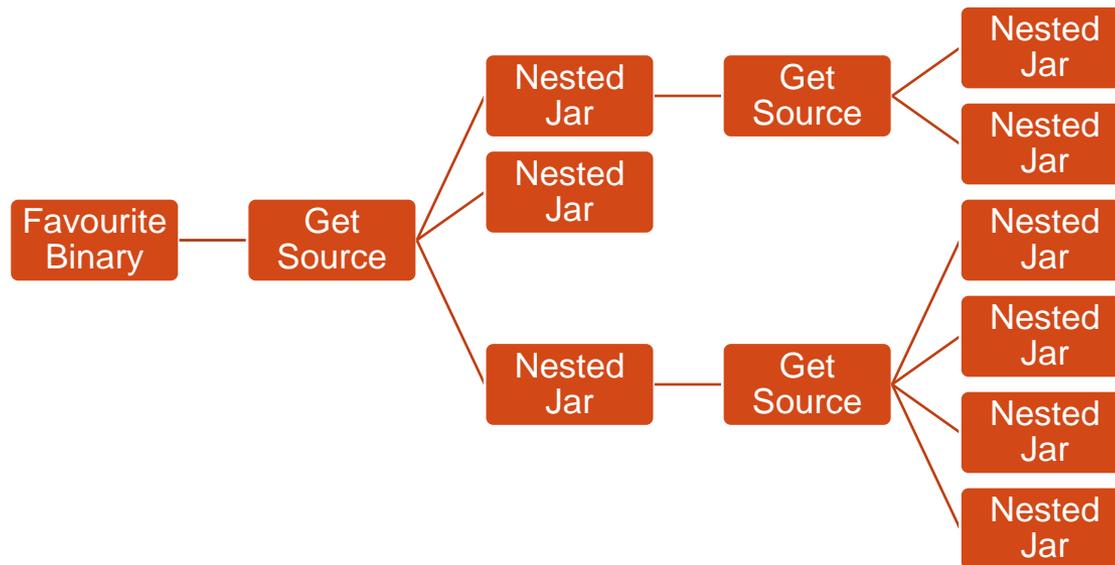
Use of Third Party Code

- Does the use of the non-EPL content introduce unnecessary risk?
- This Risk can be Mitigated by Leveraging Opportunities for Re-Use



Third Party Packages

- It is common for more material to be included in the distribution than is needed.
- Can we narrow the scope?



Example – Apache Muse 2.0

- Committer wanted to use Muse 2.0
- Grabs the binary – one file – set to go....
- Reads up on Muse 2.0 and finds that some of the functionality is dependent on Axis 2 Version 1.1.
- Committer grabs another binary.
- And now we have two – this is going to be easy

Muse 2.0 – 1st Level of Nesting

muse-2.0.0-bin\modules\axis2\muse-platform-axis2-2.0.0.jar
muse-2.0.0-bin\modules\core\muse-core-2.0.0.jar
muse-2.0.0-bin\modules\core\muse-util-2.0.0.jar
muse-2.0.0-bin\modules\core\muse-util-qname-2.0.0.jar
muse-2.0.0-bin\modules\core\muse-util-xml-2.0.0.jar
muse-2.0.0-bin\modules\core\muse-util-xstream-2.0.0.jar
muse-2.0.0-bin\modules\core\muse-wsa-soap-2.0.0.jar
muse-2.0.0-bin\modules\osgi\core\muse-core-2.0.0.jar
muse-2.0.0-bin\modules\osgi\core\muse-osgi-core-2.0.0.jar
muse-2.0.0-bin\modules\osgi\core\muse-osgi-soa-axis2-2.0.0.jar
muse-2.0.0-bin\modules\osgi\core\muse-osgi-soa-core-2.0.0.jar
muse-2.0.0-bin\modules\osgi\core\muse-util-all-2.0.0.jar
muse-2.0.0-bin\modules\osgi\core\muse-wsa-soap-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-api\muse-wsdm-muws-adv-api-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-api\muse-wsdm-muws-api-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-api\muse-wsdm-wef-api-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-api\muse-wsn-api-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-api\muse-wsrf-api-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-api\muse-wsx-api-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-impl\muse-wsdm-muws-adv-impl-2.0.0.jar

muse-2.0.0-bin\modules\osgi\ws-fx-impl\muse-wsdm-muws-impl-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-impl\muse-wsdm-wef-impl-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-impl\muse-wsn-impl-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-impl\muse-wsrf-impl-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-impl\muse-wsrf-rmd-2.0.0.jar
muse-2.0.0-bin\modules\osgi\ws-fx-impl\muse-wsx-impl-2.0.0.jar
muse-2.0.0-bin\modules\tools\muse-tools-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-api\muse-wsdm-muws-adv-api-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-api\muse-wsdm-muws-api-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-api\muse-wsdm-wef-api-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-api\muse-wsn-api-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-api\muse-wsrf-api-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-api\muse-wsx-api-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-impl\muse-wsdm-muws-adv-impl-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-impl\muse-wsdm-muws-impl-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-impl\muse-wsdm-wef-impl-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-impl\muse-wsn-impl-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-impl\muse-wsrf-impl-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-impl\muse-wsrf-rmd-2.0.0.jar
muse-2.0.0-bin\modules\ws-fx-impl\muse-wsx-impl-2.0.0.jar

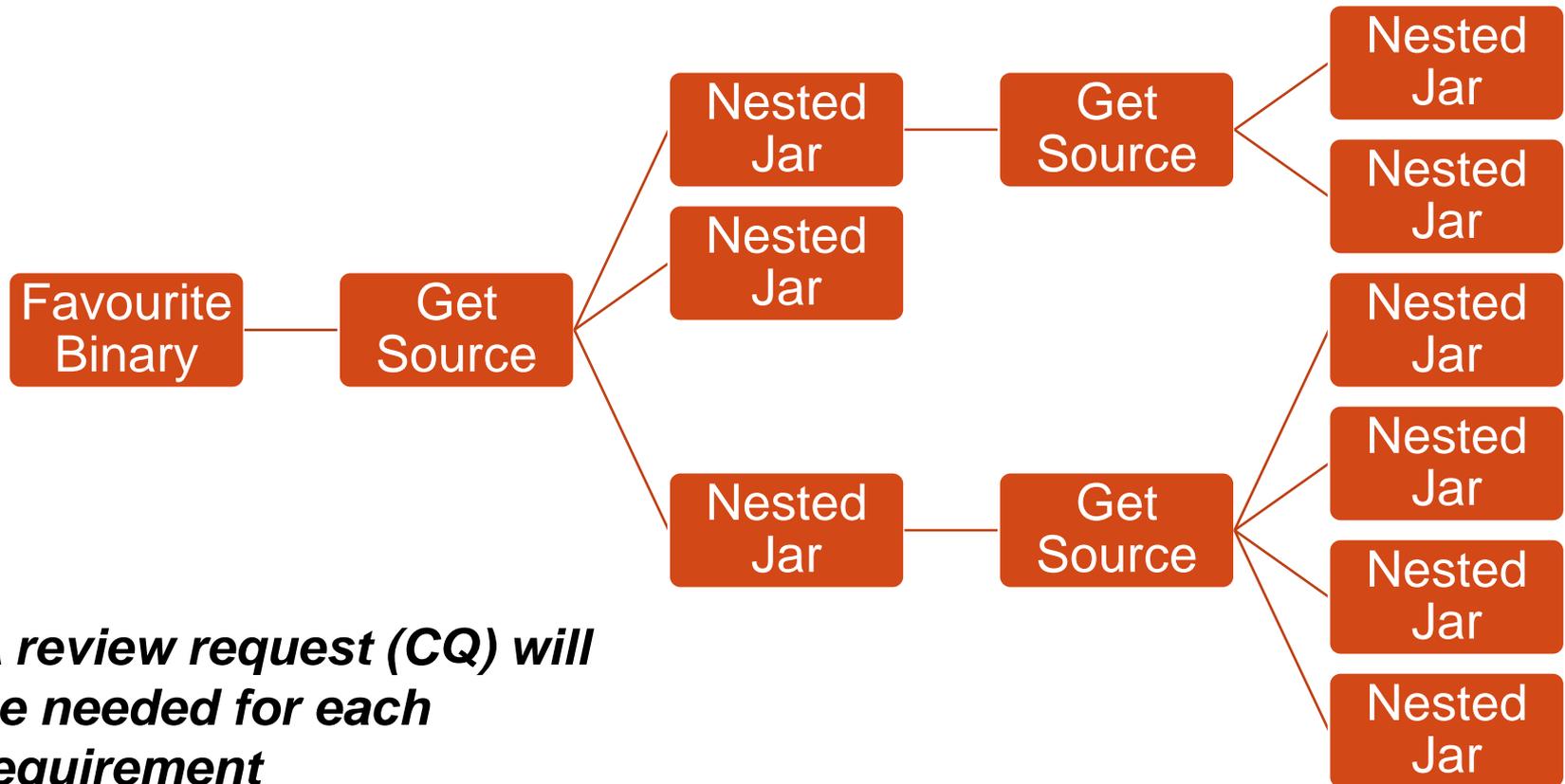
<http://www.apache.org/dist/ws/muse/2.0.0/bin>

Axis 2 v. 1.1 – 1st Level of Nesting

```
\axis2-1.1\lib\activation-1.1.jar
\axis2-1.1\lib\axiom-api-1.2.jar
\axis2-1.1\lib\axiom-impl-1.2.jar
\axis2-1.1\lib\axis2-adb-codegen-1.1.jar
\axis2-1.1\lib\axis2-java2wsdl-1.1.jar
\axis2-1.1\lib\axis2-kernel-1.1.jar
\axis2-1.1\lib\axis2-soapmonitor-1.1.jar
\axis2-1.1\lib\axis2-tools-1.1.jar
\axis2-1.1\lib\backport-util-concurrent-2.2.jar
\axis2-1.1\lib\commons-fileupload-1.1.1.jar
\axis2-1.1\lib\commons-io-1.2.jar
\axis2-1.1\lib\geronimo-spec-jms-1.1-rc4.jar
\axis2-1.1\lib\jaxen-1.1-beta-10.jar
\axis2-1.1\lib\jibx-run-1.1.2.jar
\axis2-1.1\lib\neethi-2.0.jar
\axis2-1.1\lib\stax-api-1.0.1.jar
\axis2-1.1\lib\wsdl4j-1.6.1.jar
\axis2-1.1\lib\xalan-2.7.0.jar
\axis2-1.1\lib\xercesImpl-2.8.1.jar
\axis2-1.1\lib\XmlSchema-1.2.jar
\axis2-1.1\lib\annogen-0.1.0.jar
\axis2-1.1\lib\axiom-dom-1.2.jar
\axis2-1.1\lib\axis2-adb-1.1.jar
\axis2-1.1\lib\axis2-codegen-1.1.jar
\axis2-1.1\lib\axis2-jibx-1.1.jar
\axis2-1.1\lib\axis2-saaj-1.1.jar
\axis2-1.1\lib\axis2-spring-1.1.jar
\axis2-1.1\lib\axis2-xmlbeans-1.1.jar
\axis2-1.1\lib\commons-codec-1.3.jar
\axis2-1.1\lib\commons-httpclient-3.0.1.jar
\axis2-1.1\lib\commons-logging-1.1.jar
\axis2-1.1\lib\jakarta-httpcore-4.0-alpha2.jar
\axis2-1.1\lib\jibx-bind-1.1.2.jar
\axis2-1.1\lib\mail-1.4.jar
\axis2-1.1\lib\servletapi-2.3.jar
\axis2-1.1\lib\woden-1.0.0M6.jar
\axis2-1.1\lib\wstx-asl-3.0.1.jar
\axis2-1.1\lib\xbean-2.2.0.jar
\axis2-1.1\lib\xml-apis-1.3.03.jar
```

http://ws.apache.org/axis2/download/1_1/download.cgi

Requirements are Identified



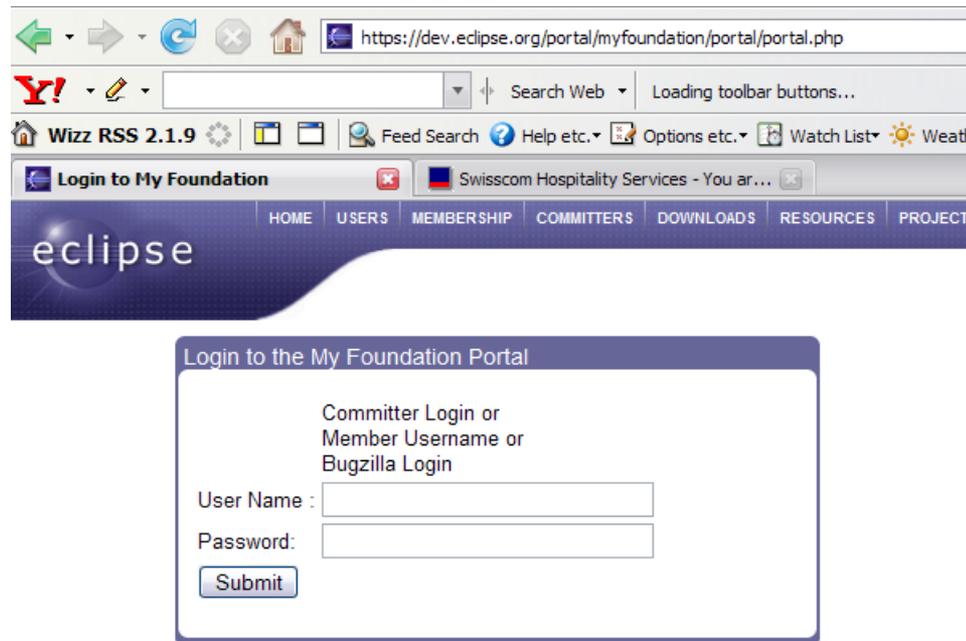
A review request (CQ) will be needed for each requirement

Review Applicable Licenses

- Determine the license(s) that you believe apply to the package.
 - These are typically re-distributed with the Package.
 - They may be found in the root directory, or in a license file.
- Are these licenses compatible?
- Uncertain? – Contact us at [license \(at\) eclipse.org](mailto:license@eclipse.org)

Submitting Review Request

- Submit Contribution Questionnaire's (CQs) for what you need via the Portal.

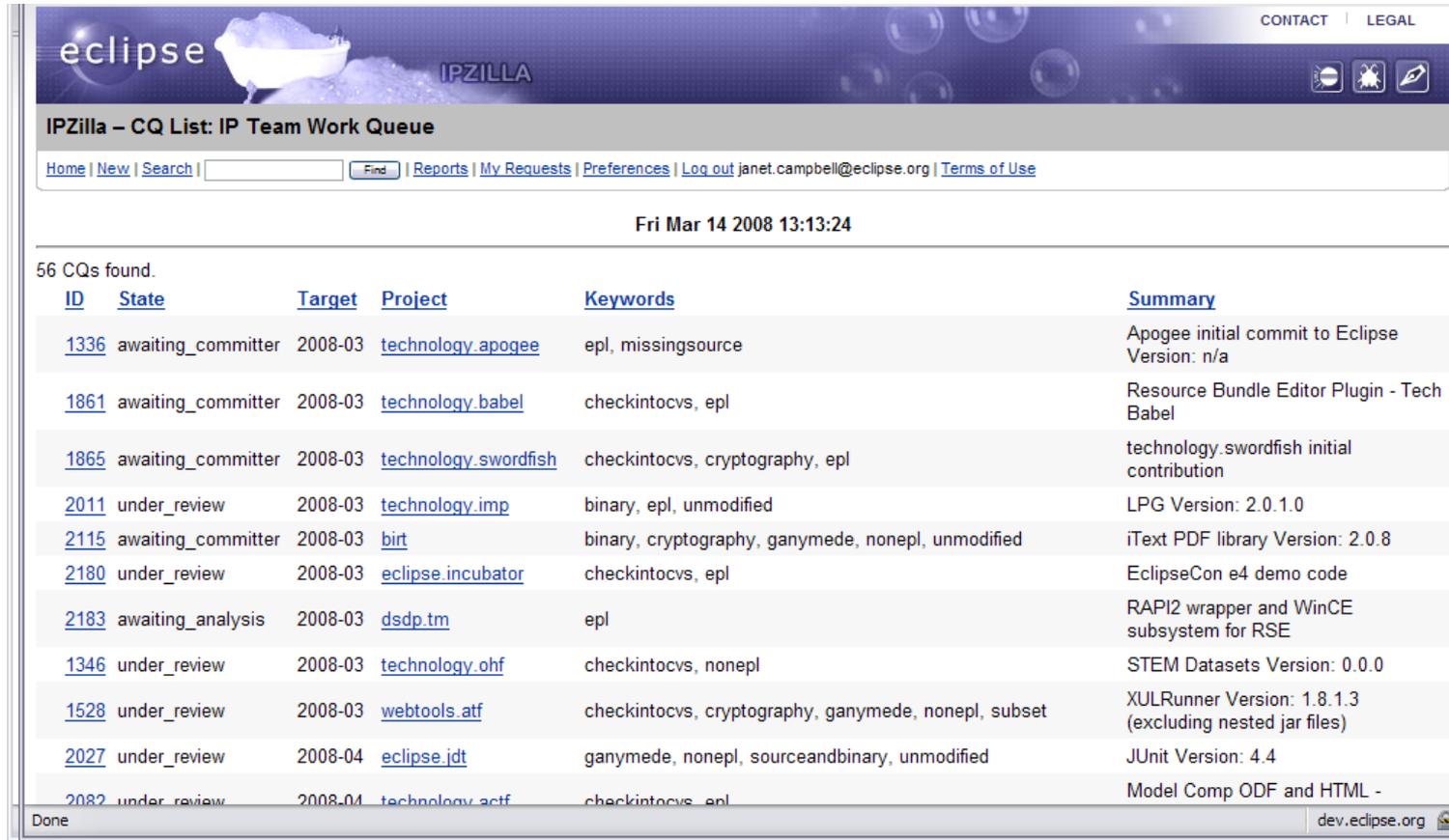


The image shows a screenshot of a web browser displaying the Eclipse Foundation Portal. The browser's address bar shows the URL `https://dev.eclipse.org/portal/myfoundation/portal/portal.php`. The page features a navigation menu with links for HOME, USERS, MEMBERSHIP, COMMITTERS, DOWNLOADS, RESOURCES, and PROJECT. Below the navigation menu is a login form titled "Login to the My Foundation Portal". The form includes a text input field for "User Name" and a text input field for "Password", both preceded by their respective labels. A "Submit" button is located below the password field. The form also includes a label for "Committer Login or Member Username or Bugzilla Login" above the user name field.

IPZilla Workflow

- Your Portal entry will create an “IPBug” in IPZilla where you will be able to monitor the progress of your request.
- The PMC must use the flag in IPZilla to +1 your contribution.
- Any discussion about the contribution, approval/rejection will be noted on the bug.

IP Team Work Queue – Visible via IPZilla “Canned” Searches



56 CQs found.

ID	State	Target	Project	Keywords	Summary
1336	awaiting_committer	2008-03	technology_apogee	epl, missingsource	Apogee initial commit to Eclipse Version: n/a
1861	awaiting_committer	2008-03	technology_babel	checkintocvs, epl	Resource Bundle Editor Plugin - Tech Babel
1865	awaiting_committer	2008-03	technology_swordfish	checkintocvs, cryptography, epl	technology.swordfish initial contribution
2011	under_review	2008-03	technology_imp	binary, epl, unmodified	LPG Version: 2.0.1.0
2115	awaiting_committer	2008-03	birt	binary, cryptography, ganymede, nonepl, unmodified	iText PDF library Version: 2.0.8
2180	under_review	2008-03	eclipse_incubator	checkintocvs, epl	EclipseCon e4 demo code
2183	awaiting_analysis	2008-03	dsdp_tm	epl	RAPI2 wrapper and WinCE subsystem for RSE
1346	under_review	2008-03	technology_ohf	checkintocvs, nonepl	STEM Datasets Version: 0.0.0
1528	under_review	2008-03	webtools_atf	checkintocvs, cryptography, ganymede, nonepl, subset	XULRunner Version: 1.8.1.3 (excluding nested jar files)
2027	under_review	2008-04	eclipse_jdt	ganymede, nonepl, sourceandbinary, unmodified	JUnit Version: 4.4
2082	under_review	2008-04	technology_actf	checkintocvs, epl	Model Comp ODF and HTML -

Foundation Review

The Level of Review Depends on the Source of the Contribution

1. Contributions from Eclipse Committers
2. Contributions from Contributors
3. Contributions from third party sources

Securing the Necessary Rights

Committer Contributions

- Legal Agreements are entered into to secure the necessary rights to have the code included in Eclipse.
 - Member Committer Agreement
 - Individual Committer Agreement
 - If the Individual Committer is Employed – An Employer Consent Form
- Through these Agreements, the Committer agrees that the Eclipse Public License (EPL) governs the code submitted by the Committer.

Securing the Necessary Rights

Contributor Contributions

- All content must be submitted through any of the channels existing on the Eclipse Foundation website such as, the Bugzilla bug reporting system.
- This material is licensed to others under the terms of the Eclipse Foundation Terms of Use.
- The Eclipse Foundation Terms of Use define the license terms that apply to any intellectual property submitted to the Eclipse Foundation website.
 - Modifications to EPL code are governed by the EPL
 - Modifications to code governed by another license are governed by that other license and the EPL.

Securing the Necessary Rights

Contributor Contributions continued...

- For all other contributions...
 - “you grant (or warrant that the owner of such rights has expressly granted) the Eclipse Foundation, the Members and the users of this Web-site a worldwide, unrestricted, royalty free, fully paid up, irrevocable, perpetual, non-exclusive license to use, make, reproduce, prepare derivative works of, publicly display, publicly perform, transmit, sell, distribute, sublicense or otherwise transfer such Materials, and/or derivative works thereof, and authorize third parties to do any, some or all of the foregoing including, but not limited to, sublicensing others to do any some or all of the foregoing indefinitely.”

Third Party Contributions

- Third party contributions such as code originating from another open source project (e.g. www.apache.org) are licensed under the license terms that apply to that project.
- Eclipse completes due diligence on each of these packages.

Eclipse Due Diligence

- The components that are identified as needed are submitted for review.
- Each component is examined from the standpoint of:
 1. Provenance
 2. License Compatibility
- We use tools to help us

How is Provenance Managed



*Who wrote this
stuff and how
did they agree
to the license?*

ANTLR 3.0

Developers who are involved in ongoing development of ANTLR or contribute significant code, must sign and return a “Certificate of Origin” document (www.ANTLR.org).

ANTLR Project -- Developer's Certificate of Origin

From ANTLR v3 and StringTemplate onwards, all substantial and/or active contributors must sign and fax or snailmail a copy of the [ANTLR contributors certificate of origin](#) formally agree to abide by it by signing on the bottom with the date. An email address and your full name must be included. Mail or fax to:

Terence Parr
University of San Francisco
2130 Fulton St, HR532
San Francisco, CA 94117
Fax: +1 415 422 5800

One-off contributions may be made through the [feedback page](#).

ANTLR 3.0 Cont...

Smaller contributors agree to the BSD electronically

Submission certification of origin and rights

By hitting the submit button, you are warranting and representing that you have the right to release this code or other content free of any obligations to third parties and are granting Terence Parr and ANTLR project contributors, henceforth referred to as *The ANTLR Project*, a license to incorporate it into The ANTLR Project tools (such as ANTLRWorks and StringTemplate) or related works under the [BSD license](#). (For large new code submissions or major new functionality, The ANTLR Project will ask you to become an official ANTLR project contributor). You understand that The ANTLR Project may or may not incorporate your submission and you warrant and represent the following:

1. *I created this submission.* I am the author of all contributed work submitted and further warrant and represent that such work is my original creation and I have the right to license it to The ANTLR Project for release under the BSD license. I hereby grant The ANTLR Project a nonexclusive, irrevocable, royalty-free, worldwide license to reproduce, distribute, prepare derivative works, and otherwise use this contribution as part of the ANTLR project, associated documentation, books, and tools at no cost to The ANTLR Project.
2. *I have the right to submit.* This submission does not violate the rights of any person or entity and that I have legal authority over this submission and to make this certification.
3. *If I violate another's rights, liability lies with me.* I agree to defend, indemnify, and hold The ANTLR Project and ANTLR users harmless from any claim or demand, including reasonable attorney fees, made by any third party due to or arising out of my violation of these terms and conditions or my violation of the rights of another person or entity.

I have read this and do so certify

SUBMIT

CANCEL

License Suitability

Consistency with Intended Use

Objectionable Terms

Terms that Present Potential Difficulties
for Downstream Consumers

Legal Incompatibility

Digging Deeper with Tools

- The Eclipse Foundation uses tools to assist with our review.
 - Keyword search tools
 - Code print matching tools

Two Small Words... “open source”

“I started designing internet applications in [year], when I joined [**Technology Company**]..... Very soon, when working on **customer projects**, I introduced the concept of [concept]..... I began developing a small generic framework I improved the framework as I moved on, from customer to customer. “

“Until it was time to **open source** it! ... As I spread the word about this framework within [Technology Company], several ... [**other employees**]... began using it and making modifications... Another good reason [to open source it] was that I wanted to have **feedback from the open source community and wanted to get help to improve it**. Also, it was good to be able to provide to our customers a framework that would continue to evolve and be maintained even after we left the project.” [*Emphasis added.*

Paraphrased from:

<http://jakarta.apache.org/cactus/participating/contributors.html>]

What Possible Issues are Raised?

Does the Technology Company have an ownership interest in the software?

“You hereby agree to assign to the Corporation all right, title and interest in and to any and all Inventions whether or not patentable or registrable under copyright or similar statutes, made or conceived or reduced to practice or learned by you, either alone or jointly with others, during your employment, which (a) relate to methods, apparatus, designs, products, processes or devices sold, leased, used or under construction or development by the Corporation, or otherwise relate to or pertain to the actual or anticipated business, functions, operations, research or development of the Corporation, (b) utilize any physical or intellectual property owned by the Corporation, or (c) are based on any information or knowledge gained by you through your employment with the Corporation.”

What Possible Issues are Raised?

Do Technology Company's customers have an ownership interest in the software?

“Consultant acknowledges that all right, title and interest in and to any of the deliverables developed as a result of the Services (including but not limited to all patents, copyrights, trademarks and any other intellectual property rights therein) provided hereunder are and shall remain the property of Company, and all rights, title and interest therein shall vest in Company and shall be deemed a “work made for hire” within the meaning of the U.S. Copyright Act, 17 U.S.C. Section 101 et. seq. To the extent any of the deliverables are not deemed to be a “work made for hire” Consultant hereby assigns to Company all rights, title and interest to the deliverables. At the expense and request of Company, Consultant agrees to execute all documents and do all other acts necessary in order to enable Company to protect its rights in such tangible or intangible property developed or arising directly as a result of the performance of the Services.”

What Possible Issues are Raised?

Are there other authors involved and did they consent to distribute the code under the license identified?

- ⌘ Do their employers have an interest in the code?
- ⌘ Do their customers have an interest in the code?

Changes to Project License Terms

- Keywords or code print matching may suggest that the Project has changed its' license terms.
- If such a change has happened, has it been done properly?

Additional Licenses

- Additional licenses may be found at the file level which would not have been obvious without further investigation.



Copied Material

- Keywords such as “borrowed” or “stolen” may suggest copied material.
- Is it re-licensed material?
 - Did the original license allow the re-licensing?
 - Is the license compatible?
- Have the terms of the original license been complied with?

Outline

1. Due Diligence

- Why do it?
- Who does what?

2. Leveraging Parallel IP

3. Third Party Dependencies

4. Dual Licensing

- Eligibility
- Important Considerations

Parallel IP – What is it?

- The IP Team will review any proposed use from both a provenance and license compatibility standpoint early in the process;
- If (1) the submission is eligible for parallel IP (more later); (2) the provenance appears to be well documented; and (3) the license is compatible:
 - The IP bug will be marked with the keyword “checkintoCVS” which means you can check the related code into CVS/SVN while the due diligence happens in parallel.

Leveraging Parallel IP

- Consideration is automatic
- Eligibility is determined as follows:
 1. Incubating Projects are eligible to leverage parallel IP **provided** the Project is in compliance with the with the incubating logo guidelines.
 - Incubating Logo on download page
 2. Mature Projects will be able to leverage parallel IP for third party packages where we have reviewed and approved the previous version of the package and it hasn't changed a great deal since we looked at it last.

Outline

1. Due Diligence

- Why do it?
- Who does what?

2. Leveraging Parallel IP

3. Third Party Dependencies

4. Dual Licensing

- Eligibility
- Important Considerations

Board Policy

- http://www.eclipse.org/org/documents/Eclipse_Policy_and_Procedure_for_3rd_Party_Dependencies_Final.pdf
- Dependencies are categories as a “works-with” or “pre-req” dependency.
 - Works-with are not “must have” requirements
 - Pre-req are “must have” requirements
- The Project must enter a CQ for and document all works-with and pre-req dependencies in their IP Log.
- PMC makes a determination on the PMC mailing list as to the type of dependency
- If it is a pre-req dependency, the EMO will determine whether review is required (whether the pre-req is exempt or non-exempt)

Outline

1. Due Diligence

- Why do it?
- Who does what?

2. Leveraging Parallel IP

3. Third Party Dependencies

4. Dual Licensing

- Eligibility
- Important Considerations

Dual Licensing

- In order to dual-license your project you must obtain the unanimous approval of the Eclipse Foundation Board of Directors.
- Project proposals will be considered on a case-by-case basis based on need.
- Dual licensing to date has been EPL/EDL, where the EDL is a BSD template license (“permissive”, “non-copyleft”) and EPL/Apache 2.0.

Important Considerations

- Dually licensed projects must ensure that they do not “copy and paste” EPL licensed code into their dually licensed project.
- Doing so would imply a re-licensing of the EPL code under the terms of the dually-licensed project.
- This cannot be done without the consent of the original author.

Questions?