Creation Review for the Proposed EGit Project

Planned review date of May 6th 2009
Shawn Pearce
April 27th 2009

Please address comments to news://news.eclipse.org/eclipse.egit

1. Background

Traditional source code control systems have been based around a single centralized server housing all of the project's revision history and assets. Historically this has worked well for smaller projects, especially when all project contributors have access to the same computer system and are located in the same physical workplace. Distributed open source projects (and commercial entities or virtual companies) have broken this model, making it much more difficult for contributors to participate. A centralized model promotes two classes of contributors: "blessed people" (those who have access to make changes to the project and use the source code control system) and "everyone else" (the rest of the world). Moving between classes is painful for everyone involved, and in many cases moving from "everyone else" to "blessed people" requires months of proving the contributor is trustworthy enough.

Multiple distributed version control systems (DVCS) have recently developed to fully support the truly distributed nature of modern open source projects, in particular large scale efforts such as those supported by Eclipse, Mozilla, KDE, Open Solaris, and the Linux kernel. Git, Mercurial, Monotone, Bazaar and darcs are all popular implementations of the DVCS concept, however their implementation details differ significantly.

True to their name, DVCS implementations eliminate the centralized server and place all project participants on an equal footing by providing them with identical capabilities. Brand new project contributors have access to the same tools and historical data as long-term trusted contributors, making it possible for new contributors to participate on an equal footing with no risk to the overall project. Most projects using DVCS tools flow changes from newer contributors to trusted contributors, arranging a loose hierarchy of review.

DVCS tools tend to perform better then their centralized counterparts, as network latency is only a factor when exchanging batches of new revisions. In a DVCS the most commonly used operations such as recording changes, file comparison, history inspection, branch creation and branch merging are handled entirely offline using locally stored information, allowing these operations to complete as quickly as the disk buffer cache and CPU will permit. Offline operation can be very useful in distributed open source projects where not all participants have large network bandwidth with low latency to connect them to the project's central server(s). For some participants, being able to work offline may be a necessity for them to contribute, due to limited availability (or high cost) of Internet access.

Finally, the implications of adopting a distributed SCM approach can bring significant benefits, especially for large open source projects such as Eclipse, and can help lower the barrier to entry for new contributors and make source code experimentation easier.

2. Scope

The project will implement Eclipse tooling for the JGit Java implementation of Git. Specific attention will be focused on performance. The EGit plugin meta data shall be fully compatible with the meta data created by the native Git version, so both can be used on the same checkout.

Beyond the Git support implementation, the project will work towards defining new essential extensions to the Eclipse core platform Team framework to account for the specific issues and features provided by distributed version control systems.

3. Relationship with Other Eclipse Projects/Components

As a team provider EGit wants to collaborate with several other components in the Eclipse ecosystem.

During the first iterations under the Eclipse.org umbrella EGit will integrate with the existing team provider infrastructure, supporting quickdiff functionality in text editors and Git centric operations in the Team submenu of a project. Further, EGit includes import and team provider connect wizards, to support bringing code in from an external repository.

Future iterations include tight integration with Mylyn.

4. Mentors

- Chris Aniszczyk (EclipseSource)
- Andrew Overholt (Red Hat)

5. Initial Committers

Shawn Pearce (Google) [project lead]

Original founder of the JGit and EGit projects, and a primary contributor to them since their inception in March 2006. 79% of the existing JGit code was authored by Shawn. Shawn has also been an active member of the larger Git community, contributing major functionality to the canonical C implementation in addition to his efforts developing JGit and EGit. Shawn's background includes more than 7 years of experience building Java based systems.

Robin Rosenberg (Dewire AB)

Robin is an extremely active contributor to both EGit and JGit, and has been serving as a dual-maintainer with Shawn for the past two years. Robin has contributed over 25% of the existing EGit code base. Robin has a very long experience with languages like C++ and Java, as well a wide spectrum of paradigms in software development.

Mik Kersten (Tasktop)

Mik Kersten is an existing Eclipse committer on the Mylyn project, with a strong background in the Eclipse community. Last year Mik served as a committer representitive on the Eclipse

Board of Directors. Mik's interest and involvement with the EGit project is to add Mylyn support where necessary, to keep EGit connected with the larger Mylyn based ecosystem.

Gunnar Wagenknecht (AGETO)

Gunnar is a very active member of the Eclipse Community and a prolific contributor with many years of experience on numerous Eclipse projects. Gunnar is also a member of the Technology PMC, which he represents on the Eclipse Architecture Council.

Matthias Sohn (SAP AG)

Matthias works on version control systems at SAP since 2000. Since 2002 he leads the team developing SAP's Design Time Repository. Matthias' interest and involvement with the EGit project is based on the belief that it has the potential to bring development efficiency to the next level. He convinced SAP to actively engage with the EGit project.

Christian Halstrick (SAP AG)

Christian is developing version control systems (both in C++ and Java) for more than 6 years. A big portion of the core versioning algorithms of SAP's distributed version control systems were authored by him. He is the maintainer of SAP's Design Time Repository Server.

Stefan Lay (SAP AG)

Stefan is a senior developer with solid Java and Eclipse plugin development experience. Currently he is responsible for the Eclipse integration of SAP's Design Time Repository.

6. Code Contributions

The existing JGit and EGit authors (primarily Shawn Pearce and Robin Rosenberg) will offer an initial code contribution, which is currently hosted on repo.or.cz at

http://repo.or.cz/w/egit.git

The JGit Java code is currently packaged under the org.speace.jgit.* namespace.

The EGit Java code is currently packaged under the org.spearce.egit.* namespace.

These two components are currently hosted in a single Git repository.

The initial contribution would segment these into two separate repositories, and repackage them into an eclipse.org namespace:

Component	**Current Package**	**Proposed Package**	**License**
JGit	org.spearce.jgit.*	org.eclipse.jgit.*	EDL (new style BSD)
EGit	org.spearce.egit.*	org.eclipse.egit.*	EPL

6.A IP Provenance

The majority of the existing surviving code was created by Shawn Pearce, Robin Rosenburg, Marek Zawirski and Tor Arne Vestbø. Line counts as of revision 5c12302... explain the ownership as follows:

JGit (100% Pure Java implementation of Git DVCS):

% LOC	LOC[1]	Author Name
79%	70475	Shawn O. Pearce
8%	7536	Robin Rosenberg
5%	4700	Marek Zawirski
1%	1436	Dave Watson
1%	1432	Robert Harder
1%	1373	Florian Koeberle
0%	535	Charles O'Farrell
0%	356	Yann Simon
0%	283	Constantine Plotnikov
0%	232	Jonas Fonseca
0%	78	Tor Arne Vestbø
0%	59	Vasyl' Vavrychuk
0%	47	Florian Köberle
0%	39	Imran M Yousuf
0%	34	Mike Ralphson
0%	16	Roger C. Soares
0%	15	John J. Franey
0%	4	Thad Hughes
0%	2	Gilion Goudsmit
0%	1	Daniel Cheng (aka SDiZ)
0%	1	Ruth Alkema
0%	1	Dima Zavin
100%	**88655**	**total**

EGit (Eclipse Team Provider for Git):

% LOC	LOC	Author Name
28%	7697	Shawn O. Pearce
25%	6898	Robin Rosenberg
21%	5940	Marek Zawirski
12%	3395	Tor Arne Vestbø
4%	1350	Dave Watson
4%	1332	Roger C. Soares
1%	435	Yann Simon

[1] LOC = Lines of Code. Measurement is taken as an absolute count, including comments, whitespace, and empty lines.

```
    0%      167 Vasyl' Vavrychuk
    0%       56 Robin Stocker
    0%       39 Ben Konrath
    0%       21 Guilhem Bonnefille
    0%       19 Mike Ralphson
    0%       19 Charles O'Farrell
    0%       15 Ferry Huberts
    0%       14 Rob Clevenger
    0%        5 Shunichi Fuji
    0%        5 Jing Xue
    0%        4 David Watson
    0%        3 Alex Blewitt
------------------------------
100%    27414 total
```

6.B Past License Changes

Prior to revision 53a2cc3f (26 May 2008) JGit was licensed under the GPL v2.1. In 53a2cc3f the license was changed to the Eclipse Distribution License v1.0 (EDL). The EDL was selected to ensure maximal reuse of the runtime library, and ensure there is one canonical library for all Java based projects to consume and contribute back to.

Prior to revision 2baa6eb5 (7 Jun 2008) EGit was licensed under a mixture of EPL v1.0, GPL v2.1, and LGPL v2.1. In 2baa6eb5 the license was changed to the Eclipse Public License v1.0 (EPL). The EPL was selected to ensure maximal compatibility with other Eclipse projects, including supporting code reuse from other team providers already covered by the EPL, and code reuse from EGit to other similar team providers, or as-of-yet unknown technologies.

7. Interested Parties

Thus far, interest in this project has been expressed by the following individuals:

- Ismael Juma
- Neil Bartlett
- Marek Zawirski
- Cédric Brun (Obeo)
- Ketan Padegaonkar (Thoughtworks)
- Tor-Arne Vestbø
- Martin Oberhuber (Wind River)
- Markus Knauer (EclipseSource)
- Antoine Toulme
- Daniel Spiewak
- Manuel Woelker (EclipseSource)
- Markus Alexander Kuppe
- Thomas Hallgren (Cloudsmith)
- Ahti Kitsik
- Alex Blewitt

8. Developer Community

There already is a thriving developer community hosted on the Git mailing list, git@vger.kernel.org. The existing Git repository history shows 34 individuals have contributed changes to the combined JGit and EGit project history; 16 of those individuals have been sustained contributors with multiple changes. See 6.A above for a more complete listing of current contributors.

The developer community has been growing. 1 year ago only 4 contributors had sustained contributions. In the past 12 months EGit and JGit combined have picked up approximately 12 new sustained contributors.

The developer community for EGit consists of individuals who want to see high-quality team provider support for the Git VCS from within the Eclipse platform, for use by individual users through UI integrations, and for use through other technologies such as Mylyn.

The developer community for JGit is much wider, as it consists of individuals who want to see a high-quality, 100% Pure Java implementation of the Git DVCS, for reuse within any Java based technology product.

9. User Community

The EGit plugin is currently available as a nightly build download from http://www.jgit.org/ Since the introduction of that Eclipse compatible update site we have seen a steady increase in users downloading the nightly builds:

Month	**Downloads**
2008-07	14
2008-08	152
2008-09	798
2008-10	852
2008-11	1112
2008-12	1910
2009-01	2870
2009-02	4318
2009-03	6622
2009-04	6821

We expect no problems in transitioning this existing user base from the jgit.org nightly builds to an official Eclipse.org release.

The JGit library is primarily directed at developers. Notable projects embedding JGit, or looking to embed JGit in the near future include:

- EGit
- Gerrit Code Review (http://code.google.com/p/gerrit/)
- NetBeans (http://code.google.com/p/nbgit/)
- Maven2
- Hudson Continuous Integration
- Atlassian JIRA (http://confluence.atlassian.com/display/JIRAEXT/Jira+Git+Plugin)

Migrating these existing users from the org.spearce.jgit.* namespace to org.eclipse.jgit.* will require some effort for each individual project.

10. Tenative Plan

- 2009 Q2: Release 0.5, with initial contribution from 0.5
- 2009 Q3: Release 1.0 M1
- 2009 Q4: Release 1.0 M2
- 2010 Q1: Release 1.0 M3
- 2010 Q2: Release 1.0

11. Copyright Statement