

-- Draft --

Code Recommenders 0.5 Release Review

Johannes Lerch (learch@cs.tu-darmstadt.de)

Marcel Bruch (bruch@cs.tu-darmstadt.de)

Sebastian Proksch (proksch@cs.tu-darmstadt.de)

About

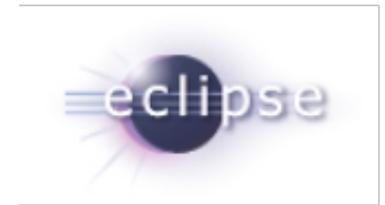


The Code Recommenders project develops tools that automatically analyze large-scale code repositories, extract interesting data from it and integrate this information back into the IDE where it is reused by developers on their daily work.

The vision of the project is to create a context-sensitive IDE that learns from what is relevant in a given code situation from its users and, in turn, give back this knowledge to other users. If you like, you may think of it like a collaborative way of sharing knowledge over the IDE

– or as „*IDE 2.0*“ (in accordance to *Web 2.0*).

Introduction



- Code Recommenders is a Technology sub-project.
 - See <http://eclipse.org/recommenders>
- This release is the 1st release at Eclipse.org.
- The goal of this release is to build an Eclipse community around it and provide the first usable version of Recommenders.



- The project is mostly staffed by Technische Universität Darmstadt (TUD) yet.
- **Committers:**
 - Andreas Kaluza (individual)
 - Eric Bodden (TUD)
 - Johannes Lerch (TUD)
 - Marcel Bruch (TUD)
 - Sebastian Proksch (TUD)
 - Stefan Henss (individual)
- Committer election for Doug Wightman, Queens University, Canada is in progress.

Contributors

So far, the project received several contributions from individuals.

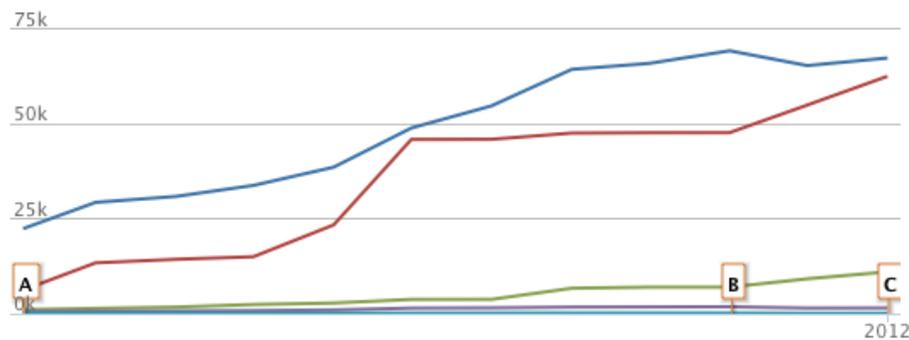
- **Contributors:**

- Andreas Frankenberg (Crowd-sourcing Server)
- Andreas Kaluza (Call Chain Completion)
- Gary Fritz (Call Chain Completion)
- Marko Martin (Call Chain Completion)
- Paul-Emmanuel Faidherbe (Subwords Completion)

Project History

Based on Ohloh.net from 01.02.2011 to 22.01.2012

- LOC Java (blue line):



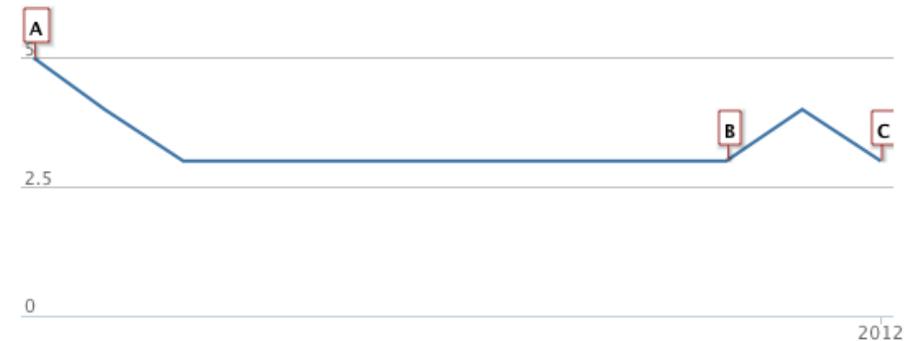
22 klocs Java code in 01.02.2011
67 klocs Java code in 22.01.2012

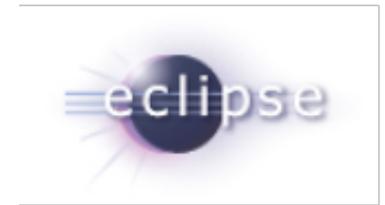
Red line represents XML. This line shows changes to .target, .pom files, etc.

- Number of commits:



- Number of active committers:





What's new in Recommenders for this release?

- **Completion Engines – completely based on JDT**
 - Intelligent Call Completion
 - Intelligent Overrides Completion
 - Naive Subwords Completion
 - Naive Chain Completion
- **Extended Documentation Platform**
 - Override Patterns Documentation Provider
 - Self-Calls Documentation Provider
 - Interactive Calls Recommender (basically code completion in documentation)
- **Zero Configuration Efforts**
 - Eclipse projects supported out-of-the-box.

Non-Code Aspects

- **User Guide:**

- In progress

- **Low activity on forum yet.**

Reasons include:

- Limited set of APIs covered → add support for Java standard library
- Limited performance → 0.5 is a big boost to eliminate this limitation
- Limited visibility → publish or perish. Good API coverage is needed before.

- **Scheduled Talks:**

- 03/2012: Eclipsecon, US
- 05/2012: Eclipse DemoCamp at FossLC.de, TU Ilmenau, DE
- 07/2012: RheinJUG Java User Group, DE

Bugzilla (as of 22.01.2012)

- Bugzilla interactions since inception:
 - 65 bugs reported
 - 49 bugs closed
 - 16 open open

IP Log



Not submitted yet.

Preliminary IP Log is available here:

- http://www.eclipse.org/projects/ip_log.php?projectid=technology.recommenders

Schedule



- **v0.5** – 30.01.2012 (Juno M6)
 - Stable and fast completion engines
 - Stable extdoc client
- **v0.6** – 19.03.2012 (Juno M6)
 - Server-side RESTful interfaces
 - Stable Snipmatch integration (tentative)
- **v0.7** – 07.05.2012 (Juno M7)
 - Recommender support for Java standard APIs
 - Stable code-search engine
 - Servers set up for Juno.
- **v1.0** – 27.06.2012 (Juno Release)
 - Community service for sharing usage data

Project Plan



- **Available at:**

- <http://eclipse.org/projects/project-plan.php?projectid=rechnology.recommenders>

- **Themes**

- Intelligent Completion Engines
 - Usability & Performance
 - Recommendation quality & API coverage
- Extended Documentation Platform
 - Improved API Coverage
- Code-search
 - Usability & Performance
- Snipmatch
 - Usability & Performance
- Building a community