Object Teams 0.7.0 Release Review

Planned Review Date: July 7, 2010

Communcation Channel: objectteams-dev@eclipse.org

Submitted by: Stephan Herrmann

Introduction

- The Object Teams Project provides tooling for developing highly modular evolvable software using the language OT/J

    - compiler and runtime for OT/J based on the JDT compiler

    - comprehensive IDE for OT/J based on the JDT

    - extension for executing OT/J code within Equinox

- Self application

    - the Object Teams Development Tooling (OTDT) extensively uses OT/Equinox for its implementation

Features

- 0.7.0 is going to be the first release of the project
- The main goal towards this release was to migrate the mature code from objectteams.org to eclipse.org
    - all IP issues have been resolved
        - changed implementation where code could not be moved to eclipse
    - code was refactored to new namespace org.eclipse...
    - build and test infrastructure was ported to eclipse servers
- User visible changes since the last release from objectteams.org (1.4.0) are summarized in:
    - http://www.eclipse.org/objectteams/pages/new_in_0.7.html

eclipse

Non-Code Aspects

- A new structure of wiki content has been created at
  - http://wiki.eclipse.org/Object_Teams
    - Object Teams Quick-Start
    - Fundamental Concepts and Technology
      - **Introduction to OT/J** — the language
      - **Object Teams Development Tooling (OTDT)** — the IDE
      - **OT/Equinox** — the module system
      - **Selected Facts** — regarding OT/J and the OTDT
    - Examples and Method
      - **Examples** — examples suitable as a further introduction to Object Teams.
      - **Patterns** — patterns for good design with OT/J
      - **Howtos** — covering both technical issues and design questions
      - **Caveats** — list of common pitfalls with explanation and solution
      - **Trouble Shooting** — error messages, causes and solution

None-Code Aspects (ctnd)

- User documentation

    - User guide is shipped with the tool

        - Intro & feature list
        - One page for each major activity
        - Language definition included
        - Extension point documentation

    - Examples are shipped as auto-installing projects

    - Some examples are documented in the wiki, too

APIs

- API of these core components are stable
  - org.eclipse.objectteams.runtime (contains the OTRE)
  - org.eclipse.objectteams.otequinox (extension point)
- Other API will be finalized before project graduation

Architectural Issues

- Already version 1.4.0 from objectteams.org had a good architecture which played well during many years of evolution

- During the migration to eclipse.org the architecture was still improved

  - avoid patched BCEL 4.4.1 but use 5.2 from Orbit

    - decision made after consulting AspectJ team who have their own patched version of BCEL.

  - drop support for JMangler framework (no longer needed)

  - revise decomposition into plug-ins in a few cases

Tool Usability

- The OTDT aims at providing the same usability and convenience for OT/J as the JDT does for Java

  - most existing features of the JDT have been extended to work with OT/J, too

  - new features have been developed specifically for OT/J:

    - views & wizards
    - code assist

- The OTDT doesn't expect the user to know every detail about Object Teams, but aims at teaching the user those bits that might be relevant to her/him.

- Quickfixes exist for migrating old OT/J projects to 0.7.0

End-of-Life

- Due to the change of namespace the OTDT 0.7.0 is **incompatible** with previous releases from objectteams.org.

- OTDT 0.7.0 supports **the same language** (OT/J) as previous releases (with some recent clarifications)

- A few plug-ins have **not yet** been migrated to eclipse

  - e.g. a metrics plug-in, IP issues not yet analyzed

- For running OT/J programs two alternative technologies had been in use (as of 1.4.0):

  - JMangler framework (**discontinued,** no longer needed)

  - Java 5 JPLIS agent (has matured by now)

Bugzilla

- 26 bugs migrated from previous Trac to Bugzilla

  - https://bugs.eclipse.org/bugs/buglist.cgi?classification=Tools;product=Objectteams; query_format=advanced;status_whiteboard_type=allwordssubstr;status_whiteboard=trac

  - 8 open bugs remain at previous Trac requiring investigation to close or migrate to Bugzilla

- 61 new bugs filed since project creation

- 58 bugs closed since project creation

  - 4 worksforme

  - 1 duplicate

  - 53 fixed

- 29 bugs currently open (all at P3)

  - 13 enhancement

  - 13 normal

  - 3 minor

Standards

- No standard exists for OT/J
  the project provides the only implementation for this technology

- OT/J is executed on a standard **Java virtual machine**

- OT/Equinox runs on the **Equinox framework** using ...

  - only standard **OSGi** headers in MANIFEST.MF

  - only regular **extensions** in plugin.xml

  - **AdapterHooks** as supported by Equinox

- Installing the OTDT (and any OT/Equinox appl.)

  - using only existing **p2** mechanisms

UI Usability

- UI closely follows the JDT lead
- localization is technically supported
    - no language packs for other languages exist yet

Schedule

- April 1: "checkin to cvs" from legal team
- Milestones
  - originally planned: for each Helios Milestone / 6 week interval
  - published in line with condensed plan:
    - April 25:   Milestone 1
    - May 7:   Milestone 2   Helios M7
    - May 31:   Milestone 3   Helios RC2 + 3 days
    - June 11:   Milestone 4   Helios RC4

Schedule (discussion & update)

- Initial plan tentatively mentioned graduation w/ Helios

  - overall time until "check in to cvs" (expectedly☺) longer than expected

    - Sept. '09:   proposal sent to EMO
    - Jan. '10:   project creation review   **> 6 months**
    - Apr. '10:   "check in to cvs"

  - many short waiting periods, single longest wait was outside Eclipse:

    - waiting for institutional provenance signatures regarding initial contribution

  - plan update after "check in to cvs"

    - too little time left to prepare graduation w/ Helios (as initially planned)
    - haven't reviewed our API - need to move some classes into "internal"

- Split maturing into

  - Release 0.7.0: now

  - Gradutation: approx. 6 months later

Communities

- Developers:

  - One committer still preparing his initial contribution for IP-analysis:

    - Olaf Otto for OT/JPA

- Users:

  - Community is not large but very qualified

    - high-detail technical discussions in the Forum
    - qualified suggestions in Bugzilla
    - great feedback at DemoCamps etc.

  - Users of the OTDT are **adopters** of OT/J & OT/Equinox

Communities (cont'd)

- Actively reaching out
  - Forum, Blog
  - Eclipse Market Place
  - Conferences
    - 4 submissions for EclipseCON'10
    - presentation at GeeCON'10
  - DemoCamps
    - Nov. '09 Berlin
    - Jun. '10 Vienna
- involvement in bugzillas throughout all of eclipse

IP Log

- IP log submitted using the tool
  - http://www.eclipse.org/projects/ip_log.php?projectid=tools.objectteams
  - Initial Contributions fully approved:
    - CQ 3784, CQ 3786, CQ 3793
  - Third-party dependencies CQs piggy-back on existing CQs
    - CQ 3805 (BCEL), CQ 3806 (XHTML DTDs)
  - Commits after initial contribution by
    - Stephan Herrmann
    - Marco Mosconi
  - No contributions from non-committers to this point

IP Issues

- The initial contribution has been reviewed and approved by Eclipse legal.

- Eclipse Software User Agreement using version of April 14, 2010 applied for all delivered features

- About files and license files are complete and correct.

Project Plan towards Graduation & beyond

- ## Contribute more components
  - OT/JPA as per the project proposal
  - metrics plugin
  - more tests
  - runtime bytecode weaver
    - prototype exists, requires testing & bug fixing
- ## Growing the community
  - reach out through all channels available
- ## Service releases every 6 weeks
  - adopt 3.6.1 when it is published
- ## Graduation roughly at the turn of the year 2010 – 2011
- ## Consider joining Indigo

Credits and Kudos

- Initial Contribution donated by
  - Technische Universität Berlin (5 authors):
    - Stephan Herrmann, Christine Hundt, Karsten Meier, Ralf Musick (previously Anklam), Jürgen Widiker
  - Fraunhofer FIRST (10 authors):
    - Jan Wloka, Carsten Pfeiffer, Andreas Kaiser, Christine Hering, Gregor Brcan, Jaroslav Svacina, Joachim Hänsel, Markus Witte, Michael Krüger, Sascha Kolewa
  - Additional individuals
    - Johannes Gebauer, Dehla Sokenou, Oliver Frank
- Guidance during establishing the project
  - Mentors, EMO, Tools PMC, Eclipse Legal
  - Special thanks to Chris Aniszczyk and Sharon Corbett for excellent feedback with very low latency