



EclipseLink 2.0.0 Release Review

Proposed Release Date: December 10th 2009

Doug Clarke, Peter Krogh - Project Leads

Project Overview



- Complete persistence framework that is both comprehensive and universal
- Supported in the following containers:
 - OSGi, Java EE, Java SE, Spring Framework
 - Equinox specific fragments included for extended usage
- Read and write objects to virtually any type of data source
 - Including Relational Databases, XML and EIS systems
- Standards Based
 - Java Persistence API (JPA): relational databases
 - Java Architecture for XML Binding (JAXB): XML
 - J2EE Connector Architecture (JCA): EIS
 - Service Data Objects (SDO)



Project Overview (cont'd)

- Milestones
 - Creation Review – May 30th, 2007
 - Initial Source Drop – Aug 23rd, 2007
 - Major releases
 - 1.0 release July 9th 2008
 - 1.1 Release: March 11th 2009
 - 1.2.0 Release: Oct 23rd 2009
 - 2.0.0 Release: Planned Dec 10th 2009
 - 2.1.0 Release: Eclipse Helios release train
 - Monthly milestone builds on multiple streams

EclipseLink 2.0.0 New Features



- JPA
 - JPA 2.0 Reference Implementation
 - Maintain JPA 1.0 compliant (since EclipseLink 1.0)
 - Fully JPA 2.0 compliant - 100% CTS pass rate
 - 174 bugs fixed in EclipseLink 2.0.0 (Foundation and JPA)
- MOXy and SDO
 - Bi-directional mapping support (MOXy)
 - 70 Bugs fixed in EclipseLink (MOXy and SDO)

Project Health



- Active Code Base
 - Average 146 svn commits / month
 - Average 2700 files / month
- Testing
 - JUnit test cases for regression testing
 - Continuous build
 - Run every half hour (if code changes)
 - Compile and Smoke tests
 - 23,000 tests run nightly on Eclipse Foundation servers
- Additional Testing
 - Additional configurations run regularly on Oracle servers
 - Several different DBs, Several Different App Servers
 - CTS testing

Project Health (cont'd)



- Documentation
 - Full Technical Docs on Wiki
 - <http://wiki.eclipse.org/EclipseLink/UserGuide>
 - Javadocs hosted online
 - <http://www.eclipse.org/eclipselink/api/latest/index.html>
 - Other documentation on wiki
 - Features index
 - FAQ
 - Examples - <http://wiki.eclipse.org/EclipseLink/Examples>
 - Over 25 How to examples
- Bugs
 - Over 250 bug/enhancements Closed in 2.0.0

Open Source



- Permeable
 - Issue discussions
 - mailing lists (eclipselink-dev and eclipselink-users)
 - Eclipse Bugzilla
 - Weekly Open Project Status Meeting
 - All code available through anonymous SVN
 - Architecture described on project Wiki along with feature design docs
- Receptive
 - Bug fixes and features implemented in a timely manner
 - Newsgroup and mailing list questions fielded promptly

Community



- Developers
 - 30 committers
 - 2 new committers from SAP
 - Code contributions from individual contributors
- Users
 - Growing community of enterprise development orgs
 - Oracle
 - Sun
 - SAP
 - TmaxSoft
 - SpringSource
 - New York City Payroll
- Distributions
 - GlassFish, Spring Framework, and Oracle

Community (cont'd)



- EclipseLink Continues to upgrade on line documentation and website to make adoption easy
 - Updated Wiki Pages
 - Javadocs available online and updated with every milestone
 - Wiki used to facilitate open communication
- Many EclipseLink presentations have been given at conferences and workshops.
 - EclipseWorld, EclipseCon, JavaOne, SpringOne, Oracle OpenWorld,
 - <http://wiki.eclipse.org/EclipseLink/Presentations>
- Many press releases have been written, and dozens of articles and blog entries have mentioned EclipseLink

API's



- Supported Spec APIs
 - Implements:
 - JPA 2.0, JPA 1.0, JAXB 2.0, JAXB 2.1, SDO 2.1, SDO 2.1.1
 - JPA 2.0 Reference Implementation (JSR 317)
 - SDO 2.1.1 Reference Implementation (JSR 235)
 - Integrates with: JDBC, JCA, JAXP, JTA
- Classic API
 - Extensive API for Persisting Java Objects
 - Migration tools and strategy from earlier Oracle TopLink releases
- Workbench
 - Swing GUI tool for design time usage
 - Metadata manipulated visually

Architectural Features



- Basic Architecture built on Product over 12 years of commercial usage
 - Stable and Performant
- Adaptable architecture - easily add/modify features in EclipseLink
 - Database Platform – add/modify support for specific database
 - Server Platform - add/modify support for specific Application Server
 - Events triggered on data store access
 - Policies – most features implemented with pluggable policies
 - Optimistic Locking
 - Caching
 - Object Instantiation
 - and many more

IP



- IP Log
 - <http://wiki.eclipse.org/EclipseLink/IPLog>
 - All IP (except 3rd Party jars) dual licensed under EPL and EDL
 - All 3rd party dependencies are IP approved for the 2.0.0
- New CQs for 2.0.0
 - Validation API – CQ 3435
 - 3rd Party contribution of JPA tests - 3552
- Community Contributions
 - 11 Bugs listed in IPLog
- Project Plan
 - <http://www.eclipse.org/projects/project-plan.php?projectid=rt.eclipselink>

End Of Life Strategy



- 1.1 release
 - No EclipseLink features are being discontinued
- 1.2.0 Release
 - Deprecation of some native persistence unit properties addressed in JPA 2.0

Schedule



- EclipseLink 2.0.0 – JPA 2.0.0 RI
 - *Start date: Aug 2008*
 - *1st Milestone: March 12th 2009*
 - Originally Planned for: May 2009
 - Original JavaEE6 release date
 - Currently planned for: Dec 10th 2009
 - Features Originally Planned:
 - Full JPA 2.0 functionality - 100% CTS passrate
 - Features Being Included
 - Full JPA 2.0 functionality - 100% CTS passrate
- Next release: EclipseLink 2.1.0
 - Helios – June 24th 2010 -
 - <http://wiki.eclipse.org/EclipseLink/Development/2.1>

Feedback



- Newsgroup:
`http://www.eclipse.org/newsportal/thread.php?group=eclipse.rt.eclipselink`
- Mailing lists:
 - eclipselink-dev@eclipse.org
 - Release Review communication channel
 - eclipselink-users@eclipse.org
- Eclipse Bugzilla
 - RT
 - Product: EclipseLink