



*Eclipse Persistence Services Project – EclipseLink
Graduation Review*

Doug Clarke, Peter Krogh - Project Leads

Project Overview



- Complete persistence framework that is both comprehensive and universal
- Will run in any Java Environment
 - OSGi, Java EE, Java SE, Spring Framework
- 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)
- Currently incubating under RT Project

Project Overview (cont'd)



- Key Contributions
 - Oracle Corp: Initial Code base came from Oracle TopLink (900K LOC)
 - 2 Project Leads + 25 Committers
 - Sun Microsystems: Contributed to initial source contribution
 - 1 committer
 - Other contributors
 - TmaxSoft
 - Adam Bien
 - 60 bugs filed by non-committers
- Milestones
 - Creation Review – May 30th, 2007
 - Initial Source Drop – Aug 23rd, 2007
 - Monthly Milestone builds – M1 Nov 5th, 2007 – M9 June 19th, 2008
 - Graduation Review – June 26th, 2008
- Project home page: <http://www.eclipse.org/eclipselink>



Project Health

- Active Code Base
 - Average 125 svn commits / month
 - Average 3000 files / month
- Testing
 - JUnit test cases for regression testing
 - Continuous build
 - Run every half hour (if code changes)
 - Compile and Smoke tests
 - 22,500 tests run nightly on Eclipse Foundation servers
- Additional Testing
 - Additional configurations run regularly on Oracle servers
 - Various DBs, Various App Servers

Project Health (cont'd)



- Documentation
 - Full Technical Docs on Wiki
 - <http://wiki.eclipse.org/EclipseLink/UserGuide>
 - Javadocs hosted online – regenerated every monthly milestone
 - <http://www.eclipse.org/eclipselink/api/1.0/index.html>
 - Other documentation on wiki
 - Features index
 - FAQ
 - Examples - <http://wiki.eclipse.org/EclipseLink/Examples>
 - Over 25 How to examples

Project Health (cont'd)



- Bugs
 - 388 bug/enhancements Closed
 - Average 45 bugs closed per month
 - Approximately 55 per month since Jan



Open Source



- Permeable
 - Issue discussions
 - mailing lists (eclipselink-dev and eclipselink-users)
 - Eclipse Bugzilla
 - Weekly Open Project Status Meeting
 - Weekly Bug council
 - 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 question fielded promptly

Open Source (cont'd)



- Intellectual Property
 - All IP (except 3rd Party jars) dual licensed under EPL and EDL
 - All 3rd party dependencies are IP approved
 - IP Log contains all CQs and all fixes from contributors.
 - <http://wiki.eclipse.org/EclipseLink/IPLog>

Community



- Developers
 - 28 committers
 - Code contributions from individual contributors
- Users
 - Growing community of enterprise development orgs
 - Oracle
 - Sun
 - TmaxSoft
 - SpringSource
 - Eclipse projects
 - EMF Teneo, others?
- Distributions
 - GlassFish, Spring Framework, and Oracle (Future)

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 presentation 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



EclipseLink 1.0 – JPA Features

- JPA 1.0 compliant implementation
- Java EE, Java SE, Web, Spring, and OSGi
- Any JDBC/SQL compliant database
- Schema generation
- Advanced features
 - Additional Mappings
 - Configurable caching
 - Extensive querying

EclipseLink 1.0 – MOXy Features



- Supports Object-XML standard - JAXB
 - Provides additional flexibility to allow complete control on how objects are mapped
- Provides complete Object-XML mapping
 - Direct, composite object, composite collection, inheritance, positional, path, transformation
 - Allows developers to work with XML as objects
 - Efficiently produce and consume XML
 - Document Preservation
- Rich set of mappings providing complete control and flexibility to map objects to any XSD
- Supports any JAXP compliant parser
 - SAX, DOM, StAX
- Visual Mapping support using Workbench

EclipseLink 1.0 – SDO Features



- SDO 2.1 Compliant
 - Marshal/Unmarshal objects to/from XML
 - Define Types/Properties programmatically or derive from XSD
 - Generate interfaces that offer JavaBean like access to DataObjects
 - Advanced mapping support for greater flexibility

EclipseLink 1.0 – EIS Features



- Provide persistence support for non-relational data stores using Java EE Connector Architecture (JCA)
- Mapping interaction inputs and outputs to persistent domain model
 - XML mapping leveraging EclipseLink MOXy and ORM mappings
 - Common Client Interface (CCI) mapping
- Visual mapping Workbench support
- Out of the box support for:
 - MQSeries, OracleAQ, Sun JCA, XML Files



- Supported Spec APIs
 - Implements: JPA 1.0, JAXB 2.1, SDO 2.1
 - 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 with 10 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



End Of Life Strategy

- 1.0 release
 - No EclipseLink features are being discontinued
- 1.1 release
 - Plan to introduce new fully functional meta data
 - 1.0 supports partial solution.
 - Will begin migration story from old meta data
 - Deprecate meta-data
 - Discontinue in a future release

Roadmap



- EclipseLink 1.0 – July, 2008
 - Delivering the full functionality of the initial contribution (Oracle TopLink)
 - Functionality Summary
 - JPA: 1.0 compliance with advanced ORM capabilities
 - MOXy: JAXB 2.1 functionality (not complete compliance) with advanced mapping capabilities
 - SDO: 2.1 compliance
 - OSGi Bundles for JPA, MOXy, and SDO components
 - Utils: Workbench and migration utilities
 - Examples: How-to's, complete examples, and tutorials
 - Documentation available on Wiki



Roadmap (continued)

- EclipseLink 1.0.1
 - Early September – *planned (1.0 + 8 weeks)*
 - Primarily a maintenance release
- EclipseLink 1.1
 - Early December – *tentative*
 - Features:
 - JPA 2.0 functionality
 - DBWS using JAX-WS
- EclipseLink 2.0
 - Spring 2009 – *tentative* – align with next Eclipse RT release?
 - Features
 - Complete JPA 2.0
 - SDO-DAS

Feedback



- Newsgroup: eclipse.technology.eclipselink
- Mailing lists:
 - eclipselink-dev@eclipse.org
 - Eclipselink-users@eclipse.org
- Eclipse Bugzilla
 - RT
 - Product: EclipseLink