Lyo OSLC4J, OAuth Library, and OSLC Test Suite 2.0.0 Release and Graduation Review

Samuel Padgett (spadgett@us.ibm.com)
Eclipse Lyo Committer

Steve Speicher (sspeiche@us.ibm.com)
Eclipse Lyo Project Lead

About

The Eclipse Lyo project is focused on providing an SDK to enable adoption of OSLC specifications. OSLC (Open Services for Lifecycle Collaboration: http://open-services.net) is an open community dedicated to reducing barriers for lifecycle tool integration. The community authors specifications for exposing lifecycle artifacts through uniform (REST) interfaces and relying on Internet and Linked Data standards.

Eclipse Lyo project is a companion to the continuing specification efforts of the OSLC community. Its main purpose is to expand adoption of OSLC specifications and to enable the Eclipse community to easily build OSLC compliant tools.

The project was approved by the PMC in July 2011 and initial contributions were committed in September 2011.

The project had a 1.0 release and graduated from incubator status on 10 October 2012. The project had a 1.1 release on 7 February 2013.

Introduction

Lyo is a Technology sub-project: http://eclipse.org/lyo

This is the third release of Lyo components as an Eclipse project. The goals are to continue to build an Eclipse community around Lyo by providing enhancements to the Java SDK for OSLC (OSLC4J), an update to the OSLC assessment test suite and code samples of OSLC provider and consumer implementations.

New in the 2.0.0 release are several enhancements:

1. An additional library to build Tracked Resource Set (TRS) implementations (org.eclipse.lyo.core.trs package)
2. A test suite for testing TRS providers (org.eclipse.lyo.testsuite.trs)
3. A W3C Linked Data Platform reference implementation (org.eclipse.lyo.ldp.*)

Early Adoption of 2.0.0

2.0.0 is an update to 1.1 with some additional enhancements. It has not undergone a formal milestone progression, but has been adopted by several Lyo consumers prior to proposing a release.

- Early adoption of 2.0.0
  - Snapshot/interim releases are being used to develop real OSLC integrations.
  - New TRS libraries and test suites have been adopted by Lyo consumers

Committer Diversity

The following individual committers are actively involved:

- Samuel Padgett (IBM)
- Michael Fiedler (IBM)
- Steve Speicher (IBM)
- Paul McMahan (IBM)

Additional active significant contributors

- David Terry
- Steven Pitschke

http://www.eclipse.org/projects/project.php?id=technology.lyo

Main Features to be released as 2.0.0

- OSLC4J – Java SDK for OSLC implementations (1.0, 1.1, 2.0.0)
  - Annotations to add OSLC attributes and meta-data to Java objects representing OSLC resources
  - Serialize/deserialize Java objects as OSLC resources
- Tracked Resource Set (TRS) SDK and reference implementation (2.0.0)
  - SDK and reference implementation for building OSLC Tracked Resource Set implementations
- OSLC assessment Test Suite (1.0, 1.1, 2.0.0)
  - Tests cover OSLC Core spec + Change Management, Quality Management, Asset Management, Requirements Management, and Automation specifications
  - (New) Coverage for TRS 2.0 specification in 2.0.0
- OAuth provider library (1.1, 2.0.0)
  - Core, persistence and webapp libraries to ease OAuth provider implementation

API Maturity

- OSLC4J API
  - No breakage of 1.0 or 1.1 API – continued stability
  - Early 2.0.0 adopters have developed real integrations – intent is to not break backward compatibility
  - Future additions to API will be designed to be non-breaking
  - Currently, no deprecated API elements

Additional Lyo project content

- Additional Lyo components NOT being formally released in 2.0.0 – under consideration for the future releases
  - SDKs for other languages (Perl)
- Lyo also contains code samples which are for developers to use as aids and examples for their own implementations. Not formally released.
  - Reference implementations for specifications
  - Sample OSLC integrations for Bugzilla, Excel and Sharepoint.
  - Client library sample implementations
- Previously proposed .NET content has moved to Microsoft Codeplex due to IP challenges with 3rd party dependencies.

Non-code aspects

OSLC4J and TestSuite documentation:

http://wiki.eclipse.org/Lyo/LyoOSLC4J
http://wiki.eclipse.org/Lyo/BuildTestSuite

TRS SDK and workshop:

http://wiki.eclipse.org/Lyo/TRSSDK
http://wiki.eclipse.org/Lyo/TRSWorkshop

LDP reference implementation:

http://wiki.eclipse.org/Lyo/BuildLDPSample

Contributor Guide:

http://wiki.eclipse.org/Lyo/ContributorsGettingStarted

Mailing List:

https://dev.eclipse.org/mailman/listinfo/lyo-dev

Forum:

http://www.eclipse.org/forums/eclipse.lyo

Continuous Integration:

Hudson-based builds

Bugzilla

Bugzilla Statistics (on 2013-09-13):

Open: 23 bugs, 58 enhancement requests

- No major, critical, or blocker bugs open

Fixed since 1.1 : 95 bugs, 27 enhancement requests

Community

- Eclipse DemoCamp : Durham, NC, November 2011
- EclipseCon Europe: Ludwigsburg, Germany, November 2011
- EmbeddedWorld: Nuremberg, Germany, February 2012
- EclipseCon: Reston, VA March 2012 – Boston, MA, March 2013
- IBM Rational Innovate: Orlando, FL, June 2012 – Orlando, FL, June 2013
- EclispeCon France: Toulouse, France, June 2013
- JavaOne: San Francisco, CA, October 2013
- OSLC Community Site: http://open-services.net
- CESAR Project: http://www.cesarproject.eu/ (Lyo adopter)

IP Log for Lyo

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

Proposed Schedule

- 1.0 – October 2012
- 1.1 – February 2013
- 2.0.0 – September 2013
  - TRS SDK, test suite, and reference implementation
  - LDP reference implementation
- 2.1.0 – 1Q 2014
  - Enhancements based on community feedback

Project Plan

http://www.eclipse.org/projects/project-plan.php?projectid=technology.lyo

http://wiki.eclipse.org/Lyo/ProjectPlans

Themes for future releases:

- Test suite improvements (additional domain coverage, improved depth of test cases, improved reporting)
- Automation enablements (Jenkins and Hudson OSLC Automation support)
- Process improvements (continuous integration)
- Improved documentation
- Usability
- Building a community