OSLC Specifications and Reference Implementation – Overview

Frédéric Loiret – KTH/OFFIS

Workshop on Life-cycle Interoperability – OSLC In Practice

Nov. 2nd 2015, Ludwigsburg
OSLC: The Executive Summary

Users can work seamlessly across their tools

- Federated
- Keep investment non-disruptive
- Integrate data instead of tools
- "Just Enough" integration
- Increased reuse
- Decreased maintenance costs
- Standard Interfaces
- Linked Data
- Architecture of the Web
- Loosely coupled
- Automation
- Monitoring
- Better visibility
- Increased traceability

OSLC User Group for Communications, 2013
Specification of the Technical Components
What to do with an OSLC spec?

How do I use it for real integrations?

- Commercial tools available with OSLC APIs
- Many organizations developing OSLC integrations in-house
- Educational and research institutions developing tool integrations based on OSLC
- Would be good to have open source sample code, SDKs, tests, etc to help get started
Eclipse Lyo

- Project kick-started in 2011 with the goal of providing an SDK to enable adoption of OSLC specifications, including
  - Code libraries
  - Reference implementations of specifications
  - Test suites and test reporting
  - Samples, tutorials and documentation
  - NOT a plugin for the Eclipse IDE, NOT related to OSGI – although Lyo artifacts could be used in Eclipse plugins.

- Project content is dual-licensed under the Eclipse Public License and the Eclipse Distribution License
Example of a new Lyo Contribution: KTH

• Automatic generation of OSLC front-ends, why?
  – To lower the threshold to develop a tool adaptor
  – Automate repetitive steps
  – Make some technologies transparent to tool developer

• Leading to …
  – A lower learning curve to adopt OSLC

• Approach: Model-based development of tool adaptors
  – Ecore meta-model to OSLC4J Java code
  – https://www.youtube.com/watch?v=nElmx_c6K5U

• Jad El-khoury (KTH), first author
  – The tool has been used by multiple projects’ partners
  – New contributions from Volvo and Thales
  – KTH is subcontracting SMEs to extend it
  – Jad became an Eclipse Lyo “contributor”, he is now a “committer”
Some Hot Questions for our workshop

• What kind of important features are currently missing in an open-source OSLC reference implementation such as Lyo?
  – Code / Technical documentation / Education / Implementation Forum / etc... ?

• How to enroll new organizations for contributing to Lyo?

• What could be the business cases for organizations (i.e., end-users, big tool providers, SMEs, academics) investing in a open community such as Lyo?