Seamless Lifecycle Integration based on Open Standards

Open Services for Lifecycle Collaboration – OSLC

? Introduction and Overview ?

Rainer Ersch
Senior Research Scientist at Siemens AG
OSLC Steering Committee Chair
rainer.ersch@siemens.com

Bola Rotibi
Research Director
Creative Intellect Consulting Ltd.
OSLC Steering Committee Member
bola@creativeintellectuk.com
Today’s products are getting more and more complex

Different engineering disciplines need to work closer together

The different roles in the engineering process are using highly specialized tools

New engineering methods and technologies have to be adopted

Legacy engineering methods have to be kept and integrated

The components for the engineering environments are provided by:

- Different vendors
- Open source projects
- In-house development
- etc. etc. etc.
The Integration Problem

- Point-to-point Integrations don’t scale
- Monocultures lock you in
- Maintenance, management, and change costs go up over time
- Ongoing and unexpected costs drain resources
- End-user productivity suffers: Either stuck with the wrong tool, stuck doing manual integration; often stuck doing both
- Integrations consume more of the IT budget: integration failures are the top 2 causes of software project delays*

- Creating new integrations is unpredictable
- Past choices restrict present action and future vision
- Even more: limited ability to respond to change
  Constrained by exhausted IT budget and lower productivity

* Commissioned study conducted by Forrester Consulting.
The Basics: What is OSLC?

Open Services for Lifecycle Collaboration (OSLC)

OSLC is a concept to link artifacts of the engineering lifecycle to achieve:

- Collaboration
- Traceability
- Reporting
- Automation

OSLC is also an open community that drives Seamless Lifecycle Integration by developing standards, providing best practice examples and a platform for users to share Lifecycle Collaboration ideas.
The Basics: why should I care?

- Tool Maker
  - create software using reusable and open assets that will interoperate with other tools both inside and outside your influence providing time and cost savings

- Tool Manager
  - reduce the complexity and risk of increasingly complex software infrastructures, and improve the value of software across a broader set of internal and external stakeholders

- Tool User
  - choose the best tools for your job and have them interact seamlessly to achieve traceability and visibility with the rest of your organization

- Systems Integrator
  - focus energy and resources on higher-value customizations, deliver more business value to your clients, and increase client satisfaction

OSLC is beneficial to many stakeholders
Contributing Organizations: http://oslc.co/organizations
Specification Technical Components

- Discovery
- HTTP C.R.U.D. for Resources
- Delegated UI for Create and Select
- UI Previews for Resource Links
- Query
- Standard Resource Representations
Specifications Technical Components

- **Create** a resource using HTTP POST and content being resource format of choice
  - URI for doing the POST is defined in the `oslc:ServiceProvider` in the `oslc:creationFactory` service

- **Read** a resource using HTTP GET and standard HTTP content negotiation
  - Client uses HTTP Accept request header to specify desired resource formats
    
  ```
  Accept: application/json, application/xml
  ```

- **Update** a resource using HTTP GET to get resource properties to be updated and HTTP PUT to send updated resource
  - Clients must preserve unknown content

- **Link** a resource using properties where values are just URIs:
  - Turtle format for a bug resource (abbrev)
  ```
  <http://example.com/bugs/2314> a oslc_cm:ChangeRequest ;
  dcterms:relation http://server/app/bugs/1235>
  ```

- **Delete** a resource using HTTP DELETE
  - Using the URI you want to delete
Specification Technical Components

Querying for resources

- Query Capability has base URI
- Clients form query URI and HTTP GET the results
- OSLC services MAY support OSLC Query Syntax
  
  \[\text{http://open-services.net/bin/view/Main/OSLCCoreSpecQuery}\]

Query syntax examples:

- Find high severity bugs created after April fool's day

  \[\text{http://example.com/bugs?oslc.where=}
  \text{cm:severity="high" and dcterms:created="2010-04-01"}\]

- Find bugs related to test case 31459

  \[\text{http://example.com/bugs?oslc.prefix=qm=}
  \text{<http://qm.example.com/ns>&oslc.where=qm:testcase=}
  \text{<http://example.com/tests/31459>}\]

- Find all bugs created by John Smith

  \[\text{http://example.com/bugs?oslc.where=}
  \text{dcterms:creator\{foaf:givenName="John" and foaf:familyName="Smith"\}}\]
Resource representations

- OSLC services should handle any type of resource
  - Not just those defined by OSLC domain specs

- Resources defined by OSLC use a RDF data model
  - therefore are simply defined by their set of properties

- OSLC services MUST produce and consume RDF/XML representations
  - Clients and services MUST NOT assume any subset of RDF/XML

- Other representations are allowed such as:
  - XML: OSLC defined format that allows for consistent formats and is RDF/XML valid
  - JSON: Rules for representing namespaces and QName properties
  - Turtle: No constraints, use as-is
  - Atom Syndication Format: `<atom:content>` SHOULD be RDF/XML
Specification Technical Components

UI Preview & Delegation

- **Rich hover:**
  Scenario supported: hover over link to get in context preview of resource

- **Resource Delegation:**
  Simple resource format defined and retrieved using HTTP content negotiation

---

## Delegated UI for Create and Select

### Rich hover:
Scenario supported: hover over link to get in context preview of resource

### Resource Delegation:
Simple resource format defined and retrieved using HTTP content negotiation
#1 Click to launch delegated UI

#2 iframe's src set to delegated UI's URL

#3 Selection made

#4 OK pressed – sends message (link+label) to parent window
Specification Technical Components

**Tracked Resource Set protocol:**

- Allows OSLC server to expose a set of resources.
- Allows OSLC clients to discover a set of resources to track additions, removals, state changes.
- Does not assume that clients will dereference the resources.
- Is suitable for dealing with large sets containing a large number of resources, as well as highly active resource sets that undergo continual change.
- Is HTTP-based and follows RESTful principles.

⇒ **Almost real-time data collection into a data warehouse which enables additional analysis, reporting and structuring of the combined data**
### OSLC Community Stepping Up

**Early State:** IBM de facto governance (with community support)

**Steering committee**
- Multi-organization steering committee established
- New governance model introduced

**OSLC at OASIS**
- Internationally recognized independent SDO
- Formal support from 22 organizations (at launch)
- Participation governed by established OASIS model

**Extend OSLC Eco System**
- Partnering with other orgs

- **2008**
- **June 2012**
- **May/June 2013**
- **Jan 2015**
OSLC Working Groups

Summary
- Core WG defines technology base
- Working with W3C Linked Data Platform WG
- 3rd major revisions in progress
- Re-scoped SCM WG to more general Config Mgmt

Participation
- Various tool provider have joined
- Leading industry companies support OSLC
- EU research projects base their Interoperability Framework on OSLC

Growing interests beyond ALM
- Integration Systems/PLM
- Integration Service Management / Cloud
- DevOps (within Automation today)

User groups
- Mobility and Embedded Systems
- ALM-PLM (2nd)
- Interoperability Patterns (coming up)

Eclipse Lyo Project
- SDKs, Reference Implementations
OSLC – Who’s involved?

Technical Committees and Working Groups are Driving Growth of Participation
Modified Agenda

- What’s new in OSLC
- What’s next in OSLC
- What’s related
- Analyst’s view
- It’s your turn:
  - Questions & Answers from the Audience to the Panel
What’s new in OSLC

OSLC Steering Committee Vision Statement

http://open-services.net/wiki/steering-committee/vision/

- Business Challenge
- Proposed Solution
  - Current Integration Capabilities
  - Envisioned Integration Capabilities
    - Integration Domains
    - Integration Toolchains
    - Federated Shared Information
- Resulting Value
- Getting Involved
What’s new in OSLC

- New OASIS Technical Committee – OSLC Domains
  - 1:1 Migration of the OSLC Domain Specs V2.0 to OASIS
  - Space for development of additional OSLC Domains
  - TC OSLC Automation will be integrated here and closed as separate TC
  - Status: Charter proposal is submitted – still open for additional supporter

  If you want to become a supporter, please send an e-mail to: webmaster@open-services.net

- OASIS OSLC Specs Public review process started
  - OSLC Core V3.0: Public Review
  - OSLC CCM Public Review starts soon
  - OSLC PROMCODE Public Review starts soon

  If you want to participate, please send an e-mail to: webmaster@open-services.net

Blog: OSLC3 Update - What is it, how is it different, and why is it important?
Most important: OSLC V3 is fully compatible with OSLC V2
# OSLC MS Affiliated TC Roadmap*

*roadmap subject to change, dependent on contributions in TC

<table>
<thead>
<tr>
<th>TC</th>
<th>Kickoff TC</th>
<th>Public Review Drafts</th>
<th>Committee Specifications</th>
<th>OASIS Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core TC</td>
<td>✓ Done</td>
<td>❐ June ’16</td>
<td>❐ July ’16</td>
<td>❐ Sept ‘16</td>
</tr>
<tr>
<td>OSLC CCM TC</td>
<td>✓ Done</td>
<td>❐ July ’16</td>
<td>❐ Aug. ‘16</td>
<td>❐ Nov ‘16</td>
</tr>
<tr>
<td>OSLC Automation TC</td>
<td>✓ Done</td>
<td>❐ TBD</td>
<td>❐ TBD</td>
<td>❐ TBD</td>
</tr>
<tr>
<td>OSLC PROMCODE TC</td>
<td>✓ Done</td>
<td>❐ July ’16</td>
<td>❐ Aug. ‘16</td>
<td>❐ Nov ‘16</td>
</tr>
<tr>
<td>OSLC RAQ Mgmt TC</td>
<td>❐ TBD</td>
<td>❐ TBD</td>
<td>❐ TBD</td>
<td>❐ TBD</td>
</tr>
</tbody>
</table>

- **Roadmap Notes:**
  - Core TC: June ‘16, July ‘16, Sept ‘16
  - OSLC CCM TC: July ‘16, Aug. ‘16, Nov ‘16
  - OSLC Automation TC: TBD
  - OSLC PROMCODE TC: July ‘16, Aug. ‘16, Nov ‘16
  - OSLC RAQ Mgmt TC: TBD

---

- Dec ‘14: Core, Change and Configuration Mgmt
- Dec ‘15: Automation, PROMCODE
- Dec ‘16: Requirements, Architecture, and Quality Mgmt

---

*roadmap subject to change, dependent on contributions in TC
What’s new in OSLC

- OSLC Survey (asking the OSLC stakeholders)

Q2: Which of the following areas are you most interested in integrating via OSLC?
Q3: Which of the following benefits would you like to achieve with lifecycle integration?
Q5: Which of the following are most likely to drive your organization to increase adoption of OSLC
Q7 Which of the following products does your organization need to integrate?

The results available at: https://www.surveymonkey.com/results/SM-2XVJT6CR/
A report will be published on http://open-services.net
If you have questions, please contact: webmaster@open-services.net
What’s next in OSLC

- **New OSLC Working Group OSLC4JS**
  

- **OSLC4JS Browser**: browsing across multiple repositories
- **OSLC4JS Client API**: a logical API abstraction layer on top of the OSLC REST services.
- **OSLC4JS Service**: JS middleware service for general and generic OSLC Domains
- **OSLC4JS Server**: generic OSLC server e.g. used with Cloud deployments
- **Status**: first implementations available on GitHub
  Usergroup kick-off meeting coming soon
What’s next in OSLC

- **New OSLC Working Group Integration Patterns**
  - Blueprints from low level integration patterns up to complex OSLC based engineering methods
    - When to link / when to sync
    - Unidirectional links vs. bi-directional links
    - Business values of OSLC based SAFe Implementations
    - ...
  - Usergroup kick-off meeting coming soon

- **OSLC Steering Committee elections**
  - Elections will be held in September
  - 2 of 9 seats will be open for election
  - (the other 7 will be open for election in Sept. 2017)
  - [http://www.oasis-oslc.org/governance](http://www.oasis-oslc.org/governance)
What’s related to OSLC

- **ALM PLM Working Group at ProSTEP iViP**
  - next level of ALM PLM Interoperability lead by Prof. Martin Eigner
  - Many new stakeholder from European Industry expected

- **IOS Coordination Forum ICF**
  - Organizational structure share the results of the European Projects regarding IOS
  - Coordination Forum to foster the enhancement of IOS standards

- **Additional Cooperation's with other organizations, tools vendors and interest groups planned**
  - Feel free to bring names of organization which might be interested to cooperate with OSLC to our attention

Contact: webmaster@open-services.net
Analyst’s view
Analyst’s view: OSLC a pillar for Interoperability

A viable strategy for the right interoperability to occur in the right way
The important areas of integration and alignment for the Software (ALM/SDLC) and Product lifecycle management (PLM) - Role perspective

### Development Management
1. Common process gates 4.5
2. Project management 4.5
3. Field problem reporting and integrated resolution 4
4. Release management 4
5. Change request handling 3.5
6. Coordination of Systems Engineering across the disciplines 3.5
7. Variation management and reuse 3.5
8. Systems and Product modelling 3
9. Integrated requirements, validation and verification 3

### Project Management
1. Integrated requirements, validation and verification 4.75
2. Change request handling 3.75
3. Systems and Product modelling 3.71
4. Coordination of Systems Engineering across the disciplines 3.43
5. Field problem reporting and integrated resolution 3.14
6. Project management 3.14
7. Variation management and reuse 3.13
8. Common process gates 3
9. Release management 2.71

### Systems Engineer
- Integrated requirements, validation and verification 4.33
- Variation management and reuse 4.17
- Change request handling 3.67
- Project management 3.67
- Coordination of Systems Engineering across the disciplines 3.5
- Release management 3.33
- Systems and Product modelling 3.33
- Field problem reporting and integrated resolution 3.17
- Common process gates 3

### Other Senior Technical Management
1. Project management 5
2. Coordination of Systems Engineering across the disciplines 4.67
3. Systems and Product modelling 4.33
4. Change request handling 4
5. Release management 4
6. Integrated requirements, validation and verification 4
7. Variation management and reuse 4
8. Common process gates 3.33
9. Field problem reporting and integrated resolution 3.33

If we look at the top 4 areas of alignment and integration between the software and product lifecycle processes for the roles which had significant representation amongst those that responded, we can see some clear variations in priorities that can be attributed to the focus areas of the different roles.
Analyst’s view: Crystal and Beyond

- Increased demand for Interoperability
- Internet of Things
- Agility, Lean, Improved Productivity & Quality
- Internet of Things
Analyst’s view: Challenges still to overcome

Silos

Overlapping and competing standards

Security
Panel Discussion

Panel Participants

- Bola Rotibi
- Rainer Ersch
OSLC ECO System

Scenario-driven & Solution-oriented

Leading choice for strategic integration technology

Generally applicable: specs available for many domains covering ALM, DevOps, ISM, and PLM

Open Services for Lifecycle Collaboration
Lifecycle integration inspired by the web

W3C® World Wide Web Consortium
Linked Data Platform Working Group

Based on and Shaping the Future of Internet Architecture

OSLC Member Section
The Resource for OSLC Implementers

OSLC: Inspired by the web
Proven
Free to use and share
Open
Changing the industry Innovative

Eclipse Lyo
Enabling tool integration with OSLC

Reference Implementations, SDK, examples

Open and Independent Governance and Leadership

OSLC Eco System