OSLC Resource Shape: A Linked Data Constraint Language

Arthur Ryman & Achille Fokoue, IBM
W3C RDF Validation Workshop, Cambridge, 2013-09-11
Topics

- About OSLC
- Resource Shape
- Use cases
- Vocabulary
- SPARQL semantics
Open Services for Lifecycle Collaboration

• “An open community building practical specifications for integrating software”
  – Change Management, Quality Management, Requirements Management, ...

• REST APIs based on Linked Data principles

• Ongoing development transitioning to:
  – W3C (Linked Data Platform)
  – OASIS
Resource Shape

• Is a high-level, declarative, programmer-friendly description of the expected contents of an RDF graph
  – RDF terms that appear in a graph typically come from many vocabularies
• Defines constraints on RDF graphs
  – unlike RDFS or OWL which define inference rules
• Does not claim to be a complete RDF constraint language
  – Covers commonly occurring cases
Use cases

• REST APIs
  – Machine-readable constraints on the RDF contained in HTTP request and response bodies
  – Suitable for programmatic validation

• Metadata for tools
  – Query builders
  – Form builders

• Documentation for humans
  – Specifications
  – Online help
Specification example

OSLC Change Management 2.0 Appendix B: Resource Shapes

An appendix of the OSLC Change Management 2.0 Specification

Overview

This defines the definitions of the OSLC CM 2.0 defined resources utilizing the OSLC Core Shape Resource definition. Implementations may use these shapes to as a basis for their work, providing their own property additions and additional constraints on OSLC defined properties.

ChangeRequest

```xml
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:dc10="http://purl.org/dc/elements/1.1/"
   xmlns:foaf="http://xmlns.com/foaf/0.1/"
   xmlns:oslc="http://open-services.net/ns/oslc#"
   xmlns:oslc_cm="http://open-services.net/ns/oslc#">
  <oslc:ResourceShape
    rdf:about="http://example.com/oslc/shapes/ChangeRequest">
    <dcterms:title>CM V3 Change Request</dcterms:title>
    <oslc:describes rdf:resource="http://open-services.net/ns/oslc#ChangeRequest"/>
    <oslc:contains rdf:resource="http://open-services.net/ns/oslc#ResourceShape"/>
  </oslc:ResourceShape>
</rdf:RDF>
```
Documentation example
Vocabulary

- Defined by [OSLC Core V2.0](#)
Vocabulary

- RDF terms are defined in Appendix A
Example

@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix oslc: <http://open-services.net/ns/core#> .
@prefix oslc_cm: <http://open-services.net/ns/cm#> .

@base <http://example.com/shape/oslc-change-request> .

<> a oslc:ResourceShape ;
   dcterms:title "Creation shape of OSLC Change Request" ;
   oslc:describes oslc_cm:ChangeRequest ;
   oslc:property <#dcterms-title>, <#oslc_cm-status> .

<#dcterms-title> a oslc:Property ;
   oslc:propertyDefinition dcterms:title ;
   oslc:occurs oslc:Exactly-one .

<#oslc_cm-status> a oslc:Property ;
   oslc:propertyDefinition oslc_cm:status ;
   oslc:occurs oslc:Zero-or-one .
### Resource: Resource Shape

A Resource Shape resource can have a title and a set of types.

- **Resource**: `osl:ResourceShape`
- **Namespace URI**: `http://open-services.net/ns/core#`
- **Suggested Namespace Prefix**: `osl`

<table>
<thead>
<tr>
<th>Prefixed Name</th>
<th>Occurs</th>
<th>Read-only</th>
<th>Value-type</th>
<th>Representation</th>
<th>Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>osl:resource</code></td>
<td>zero-or-one</td>
<td>True</td>
<td>XMLLiteral</td>
<td>n/a</td>
<td>n/a</td>
<td>Title of the resource shape. SHOULD include only content that is valid and suitable inside an XHTML <code>&lt;span&gt;</code> element.</td>
</tr>
<tr>
<td><code>osl:describes</code></td>
<td>zero-or-many</td>
<td>True</td>
<td>Resource</td>
<td>Reference</td>
<td>n/a</td>
<td>This shape describes resources that are of any of these types. Formally, a shape S applies to a resource R if there is a triple s rdf:type T and there is a triple R <code>osl:describes</code> T.</td>
</tr>
<tr>
<td><code>osl:property</code></td>
<td>zero-or-many</td>
<td>True</td>
<td>Resource</td>
<td>Inline</td>
<td><code>osl:property</code></td>
<td>The properties that are allowed or required by this shape.</td>
</tr>
</tbody>
</table>

### Resource: Allowed Values

Allowed values for one property.

- **Resource**: `osl:AllowedValues`
- **Namespace URI**: `http://open-services.net/ns/core#`
- **Suggested Namespace Prefix**: `osl`

<table>
<thead>
<tr>
<th>Prefixed Name</th>
<th>Occurs</th>
<th>Read-only</th>
<th>Value-type</th>
<th>Representation</th>
<th>Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>osl:allowedValue</code></td>
<td>one-or-many</td>
<td>True</td>
<td>Same value-type as property</td>
<td>n/a</td>
<td>n/a</td>
<td>value allowed for a property</td>
</tr>
</tbody>
</table>
**Value-type: Property**

A Property resource describes one allowed or required property of a resource.

- **Resource**: oslc:Property
- **Namespace URI**: http://open-services.net/ns/core#
- **Suggested Namespace Prefix**: oslc

<table>
<thead>
<tr>
<th>Prefixed Name</th>
<th>Occurs</th>
<th>Read-only</th>
<th>Value-type</th>
<th>Representation</th>
<th>Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dot/terms::description</td>
<td>zero-or-one</td>
<td>True</td>
<td>XMLLiteral</td>
<td>n/a</td>
<td>n/a</td>
<td>Description of the property. SHOULD include only content that is valid and suitable inside an XHTML <code>&lt;div&gt;</code> element.</td>
</tr>
<tr>
<td>dot/terms::title</td>
<td>zero-or-one</td>
<td>True</td>
<td>XMLLiteral</td>
<td>n/a</td>
<td>n/a</td>
<td>Title of the property. SHOULD include only content that is valid and suitable inside an XHTML <code>&lt;span&gt;</code> element.</td>
</tr>
<tr>
<td>oslc:allowedValues</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>oslc:AllowedValues</td>
<td>n/a</td>
<td>Resource with allowed values for the property being defined.</td>
</tr>
<tr>
<td>oslc:allowedValue</td>
<td>zero-or-many</td>
<td>True</td>
<td>Same value-type as the property being defined</td>
<td>n/a</td>
<td>n/a</td>
<td>A value allowed for property, inherited into property definition. If there are both oslc:allowedValue elements and an oslc:allowedValue resource, then the full-set of allowed values is the union of both.</td>
</tr>
<tr>
<td>oslc:defaultValue</td>
<td>zero-one</td>
<td>True</td>
<td>Same as the property being defined</td>
<td>n/a</td>
<td>n/a</td>
<td>A default value for the property.</td>
</tr>
<tr>
<td>Property</td>
<td>Multiplicity</td>
<td>Is True</td>
<td>Data Type</td>
<td>Value</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------</td>
<td>---------</td>
<td>------------</td>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>oslc:defaultValue</td>
<td>zero-or-one</td>
<td>True</td>
<td>n/a</td>
<td>n/a</td>
<td>A default value for property, if not defined in property definition</td>
<td></td>
</tr>
<tr>
<td>oslc:hidden</td>
<td>zero-or-one</td>
<td>True</td>
<td>Boolean</td>
<td>n/a</td>
<td>A hint that indicates that property MAY be hidden when presented in a user interface</td>
<td></td>
</tr>
<tr>
<td>oslc:isMemberProperty</td>
<td>zero-or-one</td>
<td>True</td>
<td>Boolean</td>
<td>n/a</td>
<td>If set to true, this indicates that the property is a membership property, as described in the Query Syntax Specification</td>
<td>Member List Patterns. This is useful when the resource whose shape is being defined is viewed as a container of other resources. For example, look at the last example in Appendix B's RDF/XML Representation Examples: Specifying the shape of a query result, where blog:comment is defined as a membership property and comment that matches the query is returned as value of that property.</td>
</tr>
<tr>
<td>oslc:name</td>
<td>exactly-one</td>
<td>True</td>
<td>String</td>
<td>n/a</td>
<td>Name of property being defined, i.e. second part of property's Prefixed Name</td>
<td></td>
</tr>
</tbody>
</table>
| oslc:maxlength    | zero-or-one   | True    | Integer    | n/a   | For String properties
<table>
<thead>
<tr>
<th>Property</th>
<th>Occurrence</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Type</th>
<th>URI Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>oslc:maxSize</td>
<td>zero-or-one</td>
<td>True</td>
<td>n/a</td>
<td>Integer</td>
<td>n/a</td>
</tr>
<tr>
<td>oslc:occurs</td>
<td>exactly-one</td>
<td>True</td>
<td>n/a</td>
<td>Resource</td>
<td>Reference</td>
</tr>
<tr>
<td>oslc:propertyDefinition</td>
<td>exactly-one</td>
<td>True</td>
<td>n/a</td>
<td>Reference</td>
<td>n/a</td>
</tr>
<tr>
<td>oslc:range</td>
<td>zero-or-many</td>
<td>True</td>
<td>n/a</td>
<td>Resource</td>
<td>Reference</td>
</tr>
</tbody>
</table>

For String properties only, specifies maximum characters allowed. If not set, then there is no maximum or maximum is specified elsewhere.

**MUST** be either
- \[http://open-services.net/ns/core#Exactly-one](http://open-services.net/ns/core#Exactly-one) (property is required),
- \[http://open-services.net/ns/core#Zero-or-one](http://open-services.net/ns/core#Zero-or-one) (property is optional),
- \[http://open-services.net/ns/core#One-or-many](http://open-services.net/ns/core#One-or-many) (property is optional), or

URI of the property whose usage is being described.

For properties with a resource value-type, Providers MAY also specify the range of possible resource types allowed, each specified by URI. The default range is \[http://open-services.net/ns/core#One-or-many](http://open-services.net/ns/core#One-or-many).
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>oslc:readOnly</td>
<td>zero-or-one</td>
<td>True</td>
<td>false, then the property is writable. Providers SHOULD declare the property read-only when changes to the value of that property will not be accepted after the resource has been created, e.g. on PUT/PATCH requests. Consumers should note that the converse does not apply. Providers MAY reject a change to the value of a writable property.</td>
</tr>
<tr>
<td>oslc:representation</td>
<td>zero-or-one</td>
<td>True</td>
<td>Resource Reference n/a</td>
</tr>
<tr>
<td>oslc:valueType</td>
<td>zero-or-many</td>
<td>True</td>
<td>Resource Reference n/a</td>
</tr>
</tbody>
</table>
### oslc:valueType
- **zero-or-many**
- **Resource**
- **Reference**
- **n/a**

If this property is omitted, then the value type is unconstrained.

**Allowed values for oslc:valueType:**

- **Literal value-types**
  - **Boolean** - [http://www.w3.org/2001/XMLSchema#boolean](http://www.w3.org/2001/XMLSchema#boolean)
  - **DateTime** - [http://www.w3.org/2001/XMLSchema#dateTime](http://www.w3.org/2001/XMLSchema#dateTime)
  - **Decimal** - [http://www.w3.org/2001/XMLSchema#decimal](http://www.w3.org/2001/XMLSchema#decimal)
  - **Double** - [http://www.w3.org/2001/XMLSchema#double](http://www.w3.org/2001/XMLSchema#double)
  - **Float** - [http://www.w3.org/2001/XMLSchema#float](http://www.w3.org/2001/XMLSchema#float)
  - **Integer** - [http://www.w3.org/2001/XMLSchema#integer](http://www.w3.org/2001/XMLSchema#integer)
  - **String** - [http://www.w3.org/2001/XMLSchema#string](http://www.w3.org/2001/XMLSchema#string)
  - **XMLLiteral** - [http://www.w3.org/1999/02/22-rdf-syntax-ns#XMLLiteral](http://www.w3.org/1999/02/22-rdf-syntax-ns#XMLLiteral)

- **Resource value-types**
  - **Resource** - [http://open-services.net/ns/core#Resource](http://open-services.net/ns/core#Resource)
  - **Local Resource** - [http://open-services.net/ns/core#LocalResource](http://open-services.net/ns/core#LocalResource)
  - **Either Resource or Local Resource** - [http://open-services.net/ns/core#AnyResource](http://open-services.net/ns/core#AnyResource)
SPARQL semantics

• It is natural to define the semantics of Resource Shape constraints using SPARQL ASK
  – Assertions should return TRUE
  – Exceptions should return FALSE
  – If a constraint is violated, it is also useful to have a SPARQL query that returns the reason

• Some constraints implicitly refer to other graphs
  – Operation post-conditions refer to the “after” graph
  – Range constraints may refer to remote resources, e.g. asserting their type
SPARQL assertion for:
olsc:occurs oslc:Zero-or-one

prefix oslc_cm: <http://open-services.net/ns/cm#>

ASK
{
    
    select ?resource
    where {
        ?resource a oslc_cm:ChangeRequest.
        ?resource oslc_cm:status ?status
    }
    group by ?resource
    having (count(?status) <= 1)
}
}
SPARQL exception for:
olsc:occurs oslc:Exactly-one

prefix oslc_cm: <http://open-services.net/ns/cm#>

ASK
{
  ?resource a oslc_cm:ChangeRequest.
  FILTER (?status1 != ?status2)
}
Summary

- OSLC Resource Shape is a high-level RDF vocabulary for describing commonly occurring constraints on RDF graphs
  - NOT a full constraint language
- Motivated by REST API documentation and tool metadata use cases (query, creation)
- Expressible as SPARQL Ask queries
- In production use at OSLC