Conversion of SysML to OSLC Resource Shapes

Axel Reichwein
October 31, 2013
OSLC Adapter for SysML Tool

- OSLC Adapter publishes OSLC resources in RDF (subject-predicate-object form)

OSLC Adapter

SysML Tool

OSLC Resources in RDF

RDF resources describing:
- SysML model elements
- OSLC services (query capabilities and creation factories)
- SysML model element types (resource shapes)
Example OSLC Resource

Subject: Requirement „Master Cylinder Efficacy“

<table>
<thead>
<tr>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>elaboratedBy</td>
<td>Use Case „Decelerate Car“</td>
</tr>
<tr>
<td>satisfiedBy</td>
<td>Block „Brake System“</td>
</tr>
<tr>
<td>derivedRqt</td>
<td>Requirement „Loss of Fluid“</td>
</tr>
<tr>
<td>derivedRqt</td>
<td>Requirement „Reservoir“</td>
</tr>
</tbody>
</table>

RDF = set of simple subject-predicate-object statements (triples)

Axel Reichwein - October 31, 2013
Standardized resource properties facilitate interoperability.

OSLC Adapter

SysML Tool

<table>
<thead>
<tr>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>elaboratedBy</td>
<td>Use Case „Decelerate Car“</td>
</tr>
<tr>
<td>satisfiedBy</td>
<td>Block „Brake System“</td>
</tr>
<tr>
<td>derivedRqt</td>
<td>Requirement „Loss of Fluid“</td>
</tr>
<tr>
<td>derivedRqt</td>
<td>Requirement „Reservoir“</td>
</tr>
</tbody>
</table>
Advantages of Standardized Resources

OSLC Adapter
SysML Tool

OSLC Resources in RDF

Web clients understand the semantics of properties of retrieved resources

OSLC Adapter
SysML Tool

OSLC Resource in RDF

OSLC tool adapters understand the semantics of properties of incoming resources (e.g. to create new elements)
Standardized Resources

Note: OSLC resources do not have to conform to a resource shape!
OSLC Resource Shape = Type of OSLC Resource

OSLC Service Provider (Tool Adapter) Publishes resources on the Web

OSLC Resource Receives resources from web clients

Example OSLC Resource in RDF

```
<rdf:Description rdf:about="http://myDomain.com/myTool/services/myModel/requirements/Req1">
  <rdf:type rdf:resource="http://open-services.net/ns/rm#Requirement"/>
  <oslc_rm:satisfiedBy rdf:resource="http://myDomain.com/myTool/services/myModel/blocks/BlockA ">
  </oslc_rm:satisfiedBy>
</rdf:Description>
```
Example OSLC Requirement Resource in RDF

```
<rdf:Description rdf:about="http://myDomain.com/myTool/services/myModel/requirements/Req1">
  <rdf:type rdf:resource="http://open-services.net/ns/rm#Requirement"/>
  <oslc_rm:satisfiedBy rdf:resource="http://myDomain.com/myTool/services/myModel/blocks/BlockA"/>
</rdf:Description>
```

Example OSLC Requirement Resource Shape in RDF

```
<oslc:ResourceShape rdf:about="https://myDomain.com/myTool/shapes/ShapeID23">
  <oslc:describes rdf:resource="http://open-services.net/ns/rm#Requirement"/>
  <oslc:property>
    <oslc:Property>
      <oslc:name>satisfiedBy</oslc:name>
      <oslc:propertyDefinition rdf:resource="http://open-service.net/ns/rm#satisfiedBy"/>
      <oslc:occurs rdf:resource="http://open-service.net/ns/core#Zero-or-many"/>
      <oslc:range rdf:resource="http://open-services.net/ns/core#Any"/>
    </oslc:Property>
  </oslc:property>
</oslc:ResourceShape>
```
OSLC Resource Shapes

• OSLC Resource Shape = RDF vocabulary that can be used for specifying and validating constraints on OSLC resources

• Various Resource Shapes have been developed for various domains (Change Request, Test Case, Requirement, Performance Monitoring Record)

• Arthur G. Ryman, Arnaud J Le Hors, Steve Speicher, *OSLC Resource Shape, A language for defining constraints on Linked Data*, Rio de Janeiro, Brazil, LDOW2013
OSLC Resource Shape

See OSLC 2.0 Appendix A Common Properties for the complete specification: [http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixA?sortcol=table;up=#oslc_ResourceShape_Resource](http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixA?sortcol=table;up=#oslc_ResourceShape_Resource)
Systems Modeling Language (SysML)

• Defined by OMG as a UML profile (http://www.omg.org/spec/SysML/1.3/)

• SysML Profile available in XMI (http://www.omg.org/spec/SysML/20120401/SysML.xmi)

• After modifications, SysML Profile also available in Ecore

• Number of concepts: 47
7.3.2.3 Rationale

Description

A Rationale documents the justification for decisions and the requirements, design, and other decisions. A Rationale can be attached to any model element including relationships. It allows the user, for example, to specify a rationale that may reference more detailed documentation such as a trade study or analysis report. Rationale is a stereotype of comment and may be attached to any other model element in the same manner as a comment.
### 7.3.9 Comment (from Kernel)

A comment is a textual annotation that can be attached to a set of elements.

**Generalizations**
- “Element (from Kernel)” on page 65.

**Attributes**
- `body: String [0..1]`
  - Specifies a string that is the comment.

**Associations**
- `annotatedElement: Element[*]`
  - References the Element(s) being commented.

### 7.3.14 Element (from Kernel)

An element is a constituent of a model. As such, it has the capability of owning other elements.

**Associations**
- `ownedComment: Comment[*]`
  - The Comments owned by this element. Subsets `Element::ownedElement`. 
SysML Rationale Resource Shape
Mapping Rules

- **Stereotype/MetaClass** => Resource Shape (only if MetaClass is not abstract)
- **Stereotype/Metaclass properties** => Resource Shape properties
  - Property name => name attribute
  - Property multiplicity => occurs attribute
  - Property type => range attribute
  - Property value type (e.g. literal or reference) => valueType attribute
- **Primitive data type** => not mapped to a resource shape. Only references to primitive data types are mapped
- **Enumeration** => not mapped to a resource shape. Only references to enumerations are mapped. If a stereotype/metaclass property type is an enumeration, the enumeration and its literals are mapped to an Allowed Values section within the resource shape describing the stereotype/metaclass
Example

- Stereotype/MetaClass => Resource Shape

```xml
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:oslc="http://open-services.net/ns/core#"
   xmlns:dcterms="http://purl.org/dc/terms/"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <dcterms:title rdf:datatype="http://www.w3.org/1999/02/22-rdf-syntax-ns#XMLLiteral">Rationale Resource Shape</dcterms:title>
  <oslc:describes rdf:resource="http://omg.org/sysml#Rationale"/>
  <oslc:property>
    <oslc:name>ownedComment</oslc:name>
    <oslc:range rdf:resource="http://omg.org/sysml#Comment"/>
    <oslc:valueType rdf:resource="http://open-services.net/ns/core#Resource"/>
    <oslc:occurs rdf:resource="http://open-service.net/ns/core#Zero-or-many"/>
  </oslc:Property>
  <oslc:property>
    <oslc:name>annotatedElement</oslc:name>
    <oslc:range rdf:resource="http://omg.org/sysml#Element"/>
    <oslc:valueType rdf:resource="http://open-services.net/ns/core#Resource"/>
    <oslc:occurs rdf:resource="http://open-service.net/ns/core#Zero-or-many"/>
  </oslc:Property>
  <oslc:property>
    <oslc:name>body</oslc:name>
    <oslc:valueType rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <oslc:occurs rdf:resource="http://open-service.net/ns/core#Zero-or-one"/>
  </oslc:Property>
</oslc:ResourceShape>
</rdf:RDF>
```
Example

- **Stereotype/Metaclass property** => Resource Shape property
  - Property name => name attribute
  - Property multiplicity => occurs attribute
  - Property type => range attribute
  - Property value type (literal or reference) => valueType attribute

```
<oslc:property>
  <oslc:Property>
    <oslc:name>body</oslc:name>
    <oslc:valueType rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <oslc:occurs rdf:resource="http://open-services.net/ns/core#Zero-or-one"/>
  </oslc:Property>
</oslc:property>
```

Attributes
- body: String [0..1]
  Specifies a string that is the comment.
Allowed Property Value Types

Allowed values for `oslc:valueType`:

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

- **Resource value-types**
  - `Resource` - http://open-services.net/ns/core#Resource
  - `Local Resource` - http://open-services.net/ns/core#LocalResource
  - `Either Resource or Local Resource` - http://open-services.net/ns/core#AnyResource

http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixA
Primitive Value Types Mapping

- SysML::PrimitiveValueTypes::String
  + UML::PrimitiveTypes::String => http://www.w3.org/2001/XMLSchema#string

- SysML::PrimitiveValueTypes::Boolean
  + UML::PrimitiveTypes::Boolean => http://www.w3.org/2001/XMLSchema#boolean

- SysML::PrimitiveValueTypes::Integer
  + UML::PrimitiveTypes::Integer => http://www.w3.org/2001/XMLSchema#integer

- **Undecided**: SysML::PrimitiveValueTypes:: Complex, SysML::PrimitiveValueTypes:: Number, SysML::PrimitiveValueTypes:: Real, UML::PrimitiveTypes::Real, UML::PrimitiveTypes:: UnlimitedNatural
Enumeration Mapping Example

Enumeration => not mapped to a resource shape. Only references to enumerations are mapped. If a stereotype/metaclass property type is an enumeration, the enumeration and its literals are mapped to an Allowed Values section within the resource shape describing the stereotype/metaclass.

```xml
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:oslc="http://open-services.net/ns/core#"
    xmlns:dcterms="http://purl.org/dc/terms/"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
    <oslc:ResourceShape rdf:about="http://myOSLCServiceProvider.com/resource/1">
        <dcterms:title rdf:datatype="http://www.w3.org/1999/02/22 rdf:resource=">
            <oslc:ResourceShape rdf:about="http://omg.org/sysml#BlockShape">
                <oslc:Property>
                    <oslc:name>isEncapsulated</oslc:name>
                    <oslc:valueType rdf:resource="http://www.w3.org/1999/02/22 rdf:resource=">
                        <oslc:occurs rdf:resource="http://open-services.net/ns/core#Zero-or-one"/>
                    </oslc:valueType>
                </oslc:Property>
            </oslc:ResourceShape>
        </dcterms:title>
        <oslc:describes rdf:resource="http://omg.org/sysml#BlockShape">
            <oslc:Property>
                <oslc:name>visibility</oslc:name>
                <oslc:allowedValues rdf:resource="http://omg.org/sysml#VisibilityKind"/>
                <oslc:occurs rdf:resource="http://open-services.net/ns/core#Zero-or-one"/>
            </oslc:Property>
        </oslc:describes>
    </oslc:ResourceShape>
</rdf:RDF>
```
Implementation Approach

- Take SysML v1.3 from OMG in XMI
- Convert SysML v1.3 to Ecore
- Copy/paste the qualified names of UML4SysML concepts (UML concepts reused in SysML) from **SysML specification** in pdf into machine-processable **text file**
- Parse SysML Profile and UML metamodel in Ecore and create corresponding OSCL Resource Shapes in RDF (Apache Jena was not used for generating RDF files because of limited formatting options. RDF files were generated as simple text files. Apache Jena is only used to test that the generated RDF files are valid RDF files which can be read as RDF files)
SysML XMI to Ecore Transformation Instructions

1. Open the original SysML.xmi in Eclipse using the UML Model Editor. (This will report “Problems encountered”, but the file actually has been opened.)

2. Save the file as SysML.uml. (Even though there are problems with the XMI file, Save As will still successfully write out a converted .uml file.)

3. Open SysML.uml in the Text Editor and search for “_StandardProfileL2_Trace”. This will identify four “generalization” elements for which the “general” subelement is missing. Add the following subelement to each of these:

   `<general xmi:type="uml:Stereotype" href="pathmap://UML_PROFILES/StandardL2.profile.uml#Trace"/>

   For example, the element

   `<generalization xmi:id="_SysML_Requirements_PackageableElement-Verify_PackageableElement-
   _generalization__Generalization._SysML_Requirements_PackageableElement-
   Verify_PackageableElement._StandardProfileL2_Trace" isSubstitutable="false"/>

   becomes

   `<generalization xmi:id="_SysML_Requirements_PackageableElement-Verify_PackageableElement-
   _generalization__Generalization._SysML_Requirements_PackageableElement-
   Verify_PackageableElement._StandardProfileL2_Trace" isSubstitutable="false">

   `<general xmi:type="uml:Stereotype" href="pathmap://UML_PROFILES/StandardL2.profile.uml#Trace"/>

   </generalization>

4. Save SysML.uml in the Text Editor and open it in the UML Model Editor. Expand the top level element and select “<Profile> SysML”. From the menu bar, choose UML Editor > Convert To > Ecore Model… to generate the equivalent Ecore model.

SysML v1.3 Profile available in XMI at http://www.omg.org/spec/SysML/20120401/SysML.xmi

Axel Reichwein - October 31, 2013
### Table 5.2 - Elements available in SysML Compliance Level 2

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Kind</th>
</tr>
</thead>
<tbody>
<tr>
<td>UML::Abstraction, UML::Action, UML::ActionExecutionSpecification, UML::Activity, UML::ActivityEdge,</td>
<td>Metaclass</td>
</tr>
<tr>
<td>UML::ActivityFinalNode, UML::ActivityGroup, UML::ActivityNode, UML::ActivityParameterNode, UML::Actor,</td>
<td></td>
</tr>
<tr>
<td>UML::AnyReceiveEvent, UML::Association, UML::Behavior, UML::BehaviorExecutionSpecification,</td>
<td></td>
</tr>
<tr>
<td>UML::BehavioralFeature, UML::BehavioredClassifier, UML::CallAction, UML::CallBehaviorAction,</td>
<td></td>
</tr>
</tbody>
</table>

### Table 5.3 - Elements available in SysML Compliance Level 3

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Kind</th>
</tr>
</thead>
<tbody>
<tr>
<td>UML::AcceptCallAction, UML::AcceptEventAction, UML::AssociationClass, UML::CreateLinkObjectAction,</td>
<td>Metaclass</td>
</tr>
<tr>
<td>UML::DataStoreNode, UML::GeneralizationSet, UML::InformationFlow, UML::InformationItem,</td>
<td></td>
</tr>
<tr>
<td>UML::InterruptibleActivityRegion, UML::Model, UML::ParameterSet, UML::ReadExtentAction,</td>
<td></td>
</tr>
<tr>
<td>UML::ReadsClassifierObjectAction, UML::ReadLinkObjectEndAction, UML::RecastObjectAction,</td>
<td></td>
</tr>
<tr>
<td>UML::ReduceAction, UML::ReclassifyClassifierBehaviorAction, UML::StartClassifierBehaviorAction,</td>
<td></td>
</tr>
<tr>
<td>UML::StartObjectBehaviorAction, UML::UnmarshalAction,</td>
<td></td>
</tr>
</tbody>
</table>

*Copy/paste (one time effort)*
Conversion Code Available on GitHub

- Code for converting SysML to OSLC Resource Shapes available in a public GitHub repository under the permissive open-source MIT license: https://github.com/axelreichwein/SysML2OSLCResourceShapes

- Main Java class for performing the conversion of SysML to OSLC ResourceShapes is ResourceShapeCreation.java: https://github.com/axelreichwein/SysML2OSLCResourceShapes/blob/master/SysMLProfileToOSLCResourceShapes/src/sysmlprofiletosoLcresourceshapes/ResourceShapeCreation.java

Generated SysML Resource Shapes

- SysMLAcceptChangeStructuralFeatureEventAction.rdf
- SysMLAllocate.rdf
- SysMLAllocateActivityPartition.rdf
- SysMLAllocate.rdf
- SysMLBindingConnector.rdf
- SysMLBlock.rdf
- SysMLChangeStructuralFeatureEvent.rdf
- SysMLConform.rdf
- SysMLConnectorProperty.rdf
- SysMLConstraintBlock.rdf
- SysMLConstraintProperty.rdf
- SysMLContinuous.rdf
- SysMLControlOperator.rdf
- SysMLCopy.rdf
- SysMLDeriveReqt.rdf
- SysMLDirectedFeature.rdf
- SysMLDiscrete.rdf
- SysMLDistributedProperty.rdf
- SysMLFlowPort.rdf
- SysMLFlowProperty.rdf
- SysMLFlowSpecification.rdf
- SysMLFullPort.rdf
- SysMLInterfaceBlock.rdf
- SysMLInvocationOnNestedPortAction.rdf
- SysMLItemFlow.rdf
- SysMLNestedConnectorEnd.rdf

- SysMLNoBuffer.rdf
- SysMLOptional.rdf
- SysMLOverwrite.rdf
- SysMLParticipantProperty.rdf
- SysMLProbability.rdf
- SysMLProblem.rdf
- SysMLPropertySpecificType.rdf
- SysMLProxyPort.rdf
- SysMLQuantityKind.rdf
- SysMLRate.rdf
- SysMLRationale.rdf
- SysMLRequirement.rdf
- SysMLRequirementRelated.rdf
- SysMLSatisfy.rdf
- SysMLTestCase.rdf
- SysMLTriggerOnNestedPort.rdf
- SysMLUnit.rdf
- SysML ValueType.rdf
- SysMLVerify.rdf
- SysMLView.rdf
- SysMLViewpoint.rdf
Generated UML4SysML (subset of UML reused for SysML) Resource Shapes

- UMLAction.rdf
- UMLActivityEdge.rdf
- UMLActivityGroup.rdf
- UMLActivityNode.rdf
- UMLBehavior.rdf
- UMLBehavioralFeature.rdf
- UMLBehavioredClassifier.rdf
- UMLCallAction.rdf
- UMLClassifier.rdf
- UMLConnectableElement.rdf
- UMLControlNode.rdf
- UMLDeployedArtifact.rdf
- UMLDeploymentTarget.rdf
- UMLDirectedRelationship.rdf
- UMLElement.rdf
- UMLEncapsulatedClassifier.rdf
- UMLEvent.rdf
- UMLExecutableNode.rdf
- UMLExecutionSpecification.rdf
- UMFLFeature.rdf
- UMLFinalNode.rdf
- UMLInteractionFragment.rdf
- UMLInvocationAction.rdf
- UMLLinkAction.rdf
- UMLLiteralSpecification.rdf
- UMLMessageEnd.rdf
- UMLMessageEvent.rdf
- UMLMultiplicityElement.rdf
- UMLNamedElement.rdf
- UMLNamespace.rdf
- UMLObjectNode.rdf
- UMLObservation.rdf
- UMLPackageableElement.rdf
- UMLParameterableElement.rdf
- UMLPin.rdf
- UMLRedefinableElement.rdf
- UMLRelationship.rdf
- UMLStructuralFeature.rdf
- UMLStructuredFeatureAction.rdf
- UMLStructuredClassifier.rdf
- UMLType.rdf
- UMLTypedElement.rdf
- UMLValueSpecification.rdf
- UMLVariableAction.rdf
- UMLVertex.rdf
- UMLWriteLinkAction.rdf
- UMLWriteStructuralFeatureAction.rdf
- UMLWriteVariableAction.rdf
SysML Block Resource Shape Example

These URIs need to be standardized.
Any Questions?

Contact me at axel.reichwein@koneksys.com