OASIS OPEN PROJECTS

OSLC Requirements Management Version 2.1. Part 2: Vocabulary

Project Specification 01
03 September 2020

This stage:

https://docs.oasis-open-projects.org/oslc-op/rm/v2.1/ps01/requirements-management-vocab.html (Authoritative)
https://docs.oasis-open-projects.org/oslc-op/rm/v2.1/ps01/requirements-management-vocab.pdf

Previous stage:

https://docs.oasis-open-projects.org/oslc-op/rm/v2.1/psd02/requirements-management-vocab.html (Authoritative)
https://docs.oasis-open-projects.org/oslc-op/rm/v2.1/psd02/requirements-management-vocab.pdf
(published as Project Specification Draft on 01 October 2019)

Latest stage:

https://docs.oasis-open-projects.org/oslc-op/rm/v2.1/requirements-management-vocab.html (Authoritative)
https://docs.oasis-open-projects.org/oslc-op/rm/v2.1/requirements-management-vocab.pdf

Latest version:

https://open-services.net/spec/rm/latest

Latest editor's draft:

https://open-services.net/spec/rm/latest-draft

Open Project:

OASIS Open Services for Lifecycle Collaboration (OSLC) OP

Project Chairs:

Jim Amsden (jamsden@us.ibm.com), IBM
Andrii Berezovskyi (andriib@kth.se), KTH

Editors:

Mark Schulte (mark.d.schulte@boeing.com), The Boeing Company
Jad El-khoury (jad@kth.se), KTH The Royal Institute of Technology

Additional components:

This specification is one component of a Work Product that also includes:

- *OSLC Requirements Management Version 2.1. Part 1: Specification*. requirements-management-spec.html
- *OSLC Requirements Management Version 2.1. Part 2: Vocabulary (this document)*. requirements-management-vocab.html

- *OSLC Requirements Management Version 2.1. Part 3: Constraints*. requirements-management-shapes.html
- *OSLC Requirements Management Version 2.1. Part 4: Machine-readable Vocabulary Terms*. requirements-management-vocab.ttl
- *OSLC Requirements Management Version 2.1. Part 5: Machine-readable Constraints*. requirements-management-shapes.ttl

Related work:

This specification is related to:

- *Open Services for Lifecycle Collaboration Requirements Management Specification Version 2.0*. http://open-services.net/bin/view/Main/RmSpecificationV2

RDF Namespaces:

http://open-services.net/ns/rm#

Abstract:

This specification defines a vocabulary for the OSLC Requirements Management domain.

Status:

This document was last revised or approved by the OASIS Open Services for Lifecycle Collaboration (OSLC) OP on the above date. The level of approval is also listed above. Check the “Latest stage” location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Open Project are listed at https://open-services.net/about/.

Comments on this work can be provided by opening issues in the project repository or by sending email to the project’s public comment list oslc-op@lists.oasis-open-projects.org.

Note that any machine-readable content (Computer Language Definitions) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product's prose narrative document(s), the content in the separate plain text file prevails.

Citation format:

When referencing this specification the following citation format should be used:
[OSLC-RM-2.1-Part2]
OSLC Requirements Management Version 2.1. Part 2: Vocabulary. Edited by Mark Schulte and Jad El-khoury. 03 September 2020. OASIS Project Specification 01. https://docs.oasis-open-projects.org/oslc-op/rm/v2.1/ps01/requirements-management-vocab.html. Latest stage: https://docs.oasis-open-projects.org/oslc-op/rm/v2.1/requirements-management-vocab.html.

Notices

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

Table of Contents

1. Introduction

This section is non-normative.

This specification defines a vocabulary for the OSLC Requirements Management resources. The intent is to define resources needed to support common integration scenarios and not to provide a comprehensive definition of a Requirement. The resource formats may not match exactly the native models supported by requirement management service providers, but are intended to be compatible with them. The approach to supporting these scenarios is to delegate operations, as driven by service provider contributed user interfaces, as much as possible and not require a service provider to expose its complete data model and application logic.

1.1 Terminology

This section is non-normative.

Terminology is based on OSLC Core Overview [OSLCCore3], W3C Linked Data Platform [LDP], W3C's Architecture of the World Wide Web [WEBARCH], Hyper-text Transfer Protocol [HTTP11]. Terminology for this specification is defined in part 1 of the multi-part specification.

1.2 References

1.2.1 Normative references

[HTTP11]

R. Fielding, Ed.; J. Reschke, Ed.. *Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing*. June 2014. Proposed Standard. URL: https://httpwg.org/specs/rfc7230.html

[LDP]

Steve Speicher; John Arwe; Ashok Malhotra. *Linked Data Platform 1.0*. 26 February 2015. W3C Recommendation. URL: https://www.w3.org/TR/ldp/

[OSLCCore3]

Jim Amsden; S. Speicher. *OSLC Core 3.0*. Project Specification. URL: https://docs.oasis-open.org/oslc-core/oslc-core/v3.0/oslc-core-v3.0-part1-overview.html

[RFC2119]

S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. March 1997. Best Current Practice. URL: https://tools.ietf.org/html/rfc2119

1.2.2 Informative references

[WEBARCH]

Ian Jacobs; Norman Walsh. *Architecture of the World Wide Web, Volume One*. 15 December 2004. W3C Recommendation. URL: https://www.w3.org/TR/webarch/

1.3 Typographical Conventions and Use of RFC Terms

As well as sections marked as non-normative, all authoring guidelines, diagrams, examples, and notes in this specification are non-normative. Everything else in this specification is normative.

The key words **MUST**, **MUST NOT**, **REQUIRED**, **SHOULD**, **SHOULD NOT**, **RECOMMENDED**, **MAY**, and **OPTIONAL** in this specification are to be interpreted as described in [RFC2119].

In addition to the namespace URIs and namespace prefixes `oslc`, `rdf`, `dcterms` and `foaf` defined in the OSLC Core specification, OSLC RM defines the namespace URI of `http://open-services.net/ns/rm#` with a namespace prefix of `oslc_rm`

2. Conformance

Requirements Management servers **MUST** use the vocabulary terms defined here where required, and with the meanings defined here.

Requirements Management servers **MAY** augment this vocabulary with additional classes, properties, and individuals.

3. Requirements Management Vocabulary Terms

This specification defines the *root* superclasses, properties and values. Servers may define additional subclasses and provide additional properties as needed.

3.1 Vocabulary Details

The namespace URI for this vocabulary is: `http://open-services.net/ns/rm#`

All vocabulary URIs defined in the OSLC Requirements Management (RM) namespace.

See Also:

- https://github.com/oslc-op/oslc-specs/blob/master/specs/rm/requirements-management-vocab.ttl

3.1.1 Classes in this namespace (2)

Requirement, RequirementCollection

Requirement

http://open-services.net/ns/rm#Requirement

Requirement is an RDFS class.

Statement of need.

RequirementCollection

http://open-services.net/ns/rm#RequirementCollection

RequirementCollection is an RDFS class.

Collection of requirements. A collection uses zero or more requirements.

3.1.2 Properties in this namespace (15)

affectedBy, constrainedBy, constrains, decomposedBy, decomposes, elaboratedBy, elaborates, implementedBy, satisfiedBy, satisfies, specifiedBy, specifies, trackedBy, uses, validatedBy

affectedBy

http://open-services.net/ns/rm#affectedBy

affectedBy is an RDF property.

Expresses an affects relationship between entities, where the object entity in some way affects the subject entity. For example, a requirement is affected by a defect.

constrainedBy

http://open-services.net/ns/rm#constrainedBy

constrainedBy is an RDF property.

Expresses a constraining relationship between entities, where the object entity constrains the subject entity. For example, a functional requirement is constrained by a safety requirement.

constrains

http://open-services.net/ns/rm#constrains

constrains is an RDF property.

Expresses a constraining relationship between entities, where the subject entity constrains the object entity. For example, a safety requirement constrains a functional requirement.

decomposedBy

http://open-services.net/ns/rm#decomposedBy

decomposedBy is an RDF property.

Expresses a decomposition relationship between entities, where the object entity decomposes the subject entity. For example, a system requirement is decomposed into a collection of system requirements.

decomposes

http://open-services.net/ns/rm#decomposes

decomposes is an RDF property.

Expresses a decomposition relationship between entities, where the subject entity decomposes the object entity. For example, a collection of system requirements decompose a system requirement.

elaboratedBy

http://open-services.net/ns/rm#elaboratedBy

elaboratedBy is an RDF property.

Expresses an elaboration relationship between entities, where the object entity elaborates the subject entity. For example, a requirement is elaborated by a model element.

elaborates

http://open-services.net/ns/rm#elaborates

elaborates is an RDF property.

Expresses an elaboration relationship between entities, where the subject entity elaborates the object entity. For example, a model element elaborates a requirement.

implementedBy

http://open-services.net/ns/rm#implementedBy

implementedBy is an RDF property.

Expresses an implementation relationship between entities, where the object entity is a necessary or desirable aspect of an implementation of the subject entity.

satisfiedBy

http://open-services.net/ns/rm#satisfiedBy

satisfiedBy is an RDF property.

The subject is satisfied by the object. For example, a user requirement is satisfied by a system requirement.

satisfies

http://open-services.net/ns/rm#satisfies

satisfies is an RDF property.

Expresses a relationship between entities, where the subject entity satisfies the object entity. For example, a system requirement satisfies a user requirement.

specifiedBy

http://open-services.net/ns/rm#specifiedBy

specifiedBy is an RDF property.

Expresses a specification relationship between entities, where the object entity further clarifies or specifies the subject entity. For example, a requirement is specified by a model element.

specifies

http://open-services.net/ns/rm#specifies

specifies is an RDF property.

Expresses a specification relationship between entities, where the subject entity further clarifies or specifies the object entity. For example, a model element specifies a requirement.

trackedBy

http://open-services.net/ns/rm#trackedBy

trackedBy is an RDF property.

Expresses a tracking relationship between entities, where the object entity in some way tracks or governs the evolution of the subject entity. For example, a requirement may be said to be tracked by a change request, in that it governs the changes to a requirement according to some process machinery.

uses

http://open-services.net/ns/rm#uses

uses is an RDF property.

Expresses a use relationship between entities, where the object entity is used by the subject entity. For example, a requirement collection may use a requirement.

validatedBy

http://open-services.net/ns/rm#validatedBy

validatedBy is an RDF property.

Expresses a validation relationship between entities, where the object entity in some way validates the subject entity. For

example, a requirement collection may be said to be validated by a test plan.

