OSLC Configuration Management Version 1.0. Part 1: Overview

OASIS Standard
23 July 2023

This stage:

https://docs.oasis-open-projects.org/oslc-op/config/v1.0/os/oslc-config-mgt.html (Authoritative)
https://docs.oasis-open-projects.org/oslc-op/config/v1.0/os/oslc-config-mgt.pdf

Previous stage:

https://docs.oasis-open-projects.org/oslc-op/config/v1.0/ps01/oslc-config-mgt.html (Authoritative)
https://docs.oasis-open-projects.org/oslc-op/config/v1.0/ps01/oslc-config-mgt.pdf

Latest stage:

https://docs.oasis-open-projects.org/oslc-op/config/v1.0/oslc-config-mgt.html (Authoritative)
https://docs.oasis-open-projects.org/oslc-op/config/v1.0/oslc-config-mgt.pdf

Latest version:

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

Latest editor's draft:

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

Open Project:

OASIS Open Services for Lifecycle Integration (OSLC) Open Project

Project Chairs:

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

Editor:

Nick Crossley (nick_crossley@us.ibm.com), IBM

Additional components:

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

- *OSLC Configuration Management Version 1.0. Part 1: Overview(this document)*. https://docs.oasis-open-projects.org/oslc-op/config/v1.0/os/oslc-config-mgt.html
- *OSLC Configuration Management Version 1.0. Part 2: Versioned Resources*. https://docs.oasis-open-projects.org/oslc-op/config/v1.0/os/versioned-resources.html
- *OSLC Configuration Management Version 1.0. Part 3: Configuration Specification*. https://docs.oasis-open-projects.org/oslc-op/config/v1.0/os/config-resources.html
- *OSLC Configuration Management Version 1.0. Part 4: RDF Vocabulary*. https://docs.oasis-open-projects.org/oslc-op/config/v1.0/os/config-vocab.html
- *OSLC Configuration Management Version 1.0. Part 5: Machine Readable Vocabulary Terms*. https://docs.oasis-open-projects.org/oslc-op/config/v1.0/os/config-vocab.ttl
- *OSLC Configuration Management Version 1.0. Part 6: Machine Readable Vocabulary Constraints*. https://docs.oasis-open-projects.org/oslc-op/config/v1.0/os/config-shapes.ttl

RDF Namespaces:

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

Abstract:

OSLC Configuration Management defines an RDF vocabulary and a set of REST APIs for managing versions and configurations of linked data resources from multiple domains.

Status:

This document was last revised or approved by the membership of OASIS 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.

The English version of this specification is the only normative version. Non-normative translations may also be available. 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-Config-1.0-Part1]
OSLC Configuration Management Version 1.0. Part 1: Overview. Edited by Nick Crossley. 23 July 2023. OASIS Standard. https://docs.oasis-open-projects.org/oslc-op/config/v1.0/os/oslc-config-mgt.html.
Latest stage: https://docs.oasis-open-projects.org/oslc-op/config/v1.0/oslc-config-mgt.html.

Copyright © OASIS Open 2023. All Rights Reserved.

Notices

Copyright © OASIS Open 2013-2023. All Rights Reserved.

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.

This specification is published under the Attribution 4.0 International (CC BY 4.0). Portions of this specification are also provided under the Apache License 2.0.

All contributions made to this project have been made under the OASIS Contributor License Agreement (CLA).

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Open Projects IPR Statements page.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Open Project or OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Project Specification or OASIS Standard, to notify the OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Open Project that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Open Project Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see https://www.oasis-open.org/policies-guidelines/trademark/ for above guidance.

Table of Contents

1. Introduction

This section is non-normative.

The specification:

- defines RDF representations of configurations and versioned resources
- defines methods for creation, updating, and deletion of aggregate (*global*) configurations
- defines how a version of a given resource is identified within the context of a configuration (global or local)
- defines how configurations local to a tool or domain contribute to a global configuration
- recommends but does not mandate creation, updating, and deletion of configurations local to a tool or domain
- allows but does not mandate capabilities for creation, deletion, and other management of versions of resources other than configurations themselves.

While the scope of this specification does not include all of [ITIL] or [ITSM], a configuration as defined by this specification shall be able to contain or reference a set of Configuration Items (CIs) and their specific versions, and hence participate in the Identification, Control, and Monitoring activities of ITIL.

1.1 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**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this specification are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.2 Terminology

This section is non-normative.

Baseline

An immutable configuration of one or more components that corresponds to some meaningful state of its resources. The set of configuration items in a baseline cannot be changed; the state of each of those items cannot be changed. Some of the metadata on the baseline itself can be changed, such as tags, comments, etc. Baselines are useful for recording deliverables, enabling a team to work with a known configuration, or as an initial state for some new stream of work.

Branch

(*verb*) to create a configuration for parallel or insulated development.
(*noun*) the result of parallel or insulated development after branching - that is, a sequence of baselines, usually with a stream at the end of the branch.

Change Set

A change set is a related set of changes, introducing new versions of resources, grouped together to ensure coherence and consistency. In some systems, a change set may be treated as a configuration that contains a delta to some other configuration; this standard allows but does not require change sets to be represented as configurations.

Component

A unit of organization consisting of a set of version resources. For a particular versioned resource, different versions can be in different components. Components are the units of configurability, and form reusable assets or building blocks. The granularity of a component varies between servers, but typically it contains the set of resources used in some product, project, or a subdivision of such a set. The resources in a component may be of any type, or multiple types, including but not limited to types defined in various OSLC domain specifications.

Concept Resource

An artifact (a requirement, test case, source code file, etc.) as an overall entity, independent of any specific version of that artifact. In Product Line Management (PLM) systems, this is often referred to as a 'master' item or part, of which there are revisions and versions or iterations.

Configuration

Copyright © OASIS Open 2023. All Rights Reserved.

A configuration identifies a set of versions of resources in a component. In some systems, those resources are called configuration items. Configurations commonly identify exactly one version of each resource in a component. Configurations can also aggregate other configurations ('contributions'), identifying further sets of versions of resources in other components, thus assembling a shared context across multiple components.

Configuration Item

A configuration item is a resource selected in a configuration - that is, it is a versioned resource. See version.

Container

A Linked Data Platform Container, often abbreviated to ***LDPC***, as defined in [LDP].

Contribution

A configuration that is a member of the set of configurations making up a configuration. The set of versioned resources in a configuration is the union of the set of versioned resources in that configuration and all its contributions; contributions are ordered to resolve any conflicts between that set.

Global Configuration

A configuration that identifies a set of configurations contributed by multiple tools or configuration management servers, allowing you to define all the relevant resources for a system. Using global configurations, you can establish a configuration context across multiple tools, even when each tool stores its resources in otherwise unrelated configurations and repositories.

Revision

In PLM systems, changes to an item are often divided into more significant changes marked as new revisions, and less significant changes marked as versions or iterations within the same revision. The boundary between revisions is often marked by a state change, such as marking the last version within a revision as 'released'. This specification does not require such a distinction between major and minor changes, using the generic term 'version' for both revision and version or iteration, but does not prohibit systems from providing the distinction.

Selections

The set of versions of resources identified by a given configuration.

Stream

A modifiable configuration of resources. For example, team members deliver to a stream when they want to make their changes visible to other team members. In many systems, changes to versioned resources can only be made in a stream.

Tag

A short string used to find, mark, qualify, or identify some resource. Resources can have multiple tags. Tags can be used to indicate the intended purpose of a resource, or to flag it for some intention.

Variant

This term is not used formally by the specifications, but informally it may refer to a version of a resource that is identified by a specific set of characteristics that distinguish it from other versions of that resource, where each variant can exist at the same time as those other versions of the resource.

Version

A referenceable state of a resource. A resource with separable referenceable versions is called a ***versioned resource***. Each version of such a resource is called a ***version resource***, and can be referenced with a distinct URI. In PLM systems, versions are often subdivided into revisions and versions or iterations.

1.3 Concept resources, version resources, and configuration context

This section is non-normative.

All the major “Artifacts” or “Entities” in OSLC domains are ***concept resources***. Examples are defects, tasks, products, physical parts or items, projects, users, tests cases requirements, plans and so on. Like all resources, concept resources have URLs. Except in a few technical and specialized scenarios, URLs of concept resources are used throughout the system. “Entities” are addressed in HTTP messages via their concept resource URLs. Links are typically established using the URL of a concept resource.

The state (including the properties) of a concept resource can vary over time, or for other reasons. A ***versioned resource*** is a concept resource that keeps track of different states. a ***version resource*** is a resource that represents one specific state of a versioned resource. Not every past state is necessarily kept; a server may elide or discard states that are no longer referenced.

Not all resource types need be versioned - for example, OSLC change requests are typically not versioned. OSLC configurations and components need not be versioned.

For a versioned resource, a GET on the URI of a concept resource normally resolves that URI to an appropriate state of (version of) that concept resource for the appropriate configuration context (see later). The returned http resource will contain RDF statements about both the version resource and the concept resource; most statements should use the concept resource as the subject, because a version resource reflects the state of (properties of) the concept resource as it appeared at some time. Using the concept resource URI as the subject of RDF statements is more consistent with the handling of non-versioned resources; using the versioned resource URI as the subject of RDF statements requires more client knowledge of versioning.

For some PLM or PLE applications, it is useful to be able to represent incomplete or abstract hierarchies as partially-bound configurations, where not all versions of the concepts in that configuration have yet been determined. That determination may be made later in time, or dynamically based on parameters defined elsewhere in the system.

Since different versions reflecting different states of the concept resource may return conflicting RDF statements about the same subject, it is important for clients handling multiple versions (multiple configurations) to keep track of the relevant versions; this can be done using RDF named graphs.

As an example, `GET https://example.com/conceptResourceA` in one configuration context might return:

EXAMPLE 1

```
:conceptResourceA-version23
    a oslc_config:VersionResource ;
    dcterms:isVersionOf :conceptResourceA . # dcterms:isVersionOf relates the version resource itself to its concept resource
:conceptResourceA
    a :someType ;
    oslc_config:versionId "23" ;
    dcterms:title "Concept Resource A" ;
    :color "blue" ;
    dcterms:description "Concept resource A as it appears in the state with the URI :conceptResourceA-version23" .
```

while in a different configuration context it might return:

EXAMPLE 2

```
:conceptResourceA-version17
    a oslc_config:VersionResource ;
    dcterms:isVersionOf :conceptResourceA .
:conceptResourceA
    a :someType ;
    oslc_config:versionId "17" ;
    dcterms:title "Concept Resource A" ;
    :color "red" ;
    dcterms:description "Concept resource A as it appears in the state with the URI :conceptResourceA-version17" .
```

To keep track of which version represented which state of the conflicting color statements, use of RDF named graphs is recommended, as shown here using [TriG]:

EXAMPLE 3

```
:conceptResourceA-version23 = {
    :conceptResourceA-version23
        a oslc_config:VersionResource ;
        dcterms:isVersionOf :conceptResourceA .
    :conceptResourceA
        a :someType ;
        oslc_config:versionId "23" ;
        dcterms:title "Concept Resource A" ;
        :color "blue" ;
        dcterms:description "Concept resource A as it appears in the state with the URI :conceptResourceA-version23" .
}.

:conceptResourceA-version17 = {
    :conceptResourceA-version17
        a oslc_config:VersionResource ;
        dcterms:isVersionOf :conceptResourceA .
    :conceptResourceA
```

```
        a :someType ;
        oslc_config:versionId "17" ;
        dcterms:title "Concept Resource A" ;
        :color "red" ;
        dcterms:description "Concept resource A as it appears in the state with the URI :conceptResourceA-version17" .
}.
```

Updating a versioned resource is typically performed in a stream, and usually results in the creation of a new version of the same concept resource. In some systems, a related batch of changes to one or more resources can be grouped into a *change set*; this version of OSLC Configuration Management does not define how change sets might work.

Since concept resource URIs for versioned resources do not inherently identify a specific version of that resource, a client needs to provide a configuration context within which the concept resource is resolved to a specific version.

A client requests a specific configuration context in a header or a query string.

1.4 References

1.4.1 Normative references

[LDP]

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

[RFC2119]

S. Bradner. *Key words for use in RFCs to Indicate Requirement Levels*. IETF, March 1997. Best Current Practice. URL: https://www.rfc-editor.org/rfc/rfc2119

[RFC8174]

B. Leiba. *Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words*. IETF, May 2017. Best Current Practice. URL: https://www.rfc-editor.org/rfc/rfc8174

1.4.2 Informative references

[ITIL]

Information Technology Infrastructure Library. URL: https://en.wikipedia.org/wiki/Information_Technology_Infrastructure_Library

[ITSM]

IT Service Management. URL: https://en.wikipedia.org/wiki/IT_service_management

[TriG]

Gavin Carothers; Andy Seaborne. *RDF 1.1 TriG*. W3C, 25 February 2014. W3C Recommendation. URL: https://www.w3.org/TR/trig/

2. RDF Namespaces

OSLC Configuration Management defines the namespace URI of `http://open-services.net/ns/config#` with a namespace prefix of `oslc_config`.

OSLC Configuration Management uses the following prefixes:

Prefix	Namespace
acc	http://open-services.net/ns/core/acc#
dcterms	http://purl.org/dc/terms/
foaf	http://xmlns.com/foaf/0.1/
ldp	http://www.w3.org/ns/ldp#
oslc	http://open-services.net/ns/core#
oslc_auto	http://open-services.net/ns/auto#
oslc_config	http://open-services.net/ns/config#
owl	http://www.w3.org/2002/07/owl#
prov	http://www.w3.org/ns/prov#
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	http://www.w3.org/2000/01/rdf-schema#
vann	http://purl.org/vocab/vann/
vs	http://www.w3.org/2003/06/sw-vocab-status/ns#
xsd	http://www.w3.org/2001/XMLSchema#

Copyright © OASIS Open 2023. All Rights Reserved.

3. Non-RDF Sources

All resources described in this specification are RDF resources. Servers **SHOULD** follow the [LDP] recommendation for handling non-RDF data sources. [config-mgt-1]

Copyright © OASIS Open 2023. All Rights Reserved.

4. Conformance

Implementations of this specification need to satisfy the following conformance clauses.

Clause Number	Requirement
config-mgt-1	All resources described in this specification are RDF resources. Servers **SHOULD** follow the [LDP] recommendation for handling non-RDF data sources.

Appendix A. Acknowledgements

This section is non-normative.

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

David Honey (Persistent)
Ian Green (Persistent)
Geoff Clemm (Persistent)
Mike Loeffler (GM)
Martin Sarabura (PTC)