OSLC Asset Management 2.0 Specification

New here? Sign up! (http://open-services.net/forums/member/register)

-
-
  - Recent changes (http://open-services.net/wiki/asset-management/Special:Recentchanges)
-
- Asset Management home (http://open-services.net/wiki/asset-management)
- All pages (http://open-services.net/wiki/asset-management/Special:Titles)
- Categories (http://open-services.net/wiki/asset-management/Special:Categories)
- Random Page (http://open-services.net/wiki/asset-management/Special:Random_page)
-
-
-
- Uploaded Files (http://open-services.net/wiki/asset-management/Special:Files)
-
- RSS (http://open-services.net/wiki/asset-management/rss/)
-
- Basic syntax guide (http://stackoverflow.com/editing-help)
- Linking and categories syntax (http://expressionengine.com/user_guide/modules/wiki/wiki_syntax.html)

Sort of a ghost town here
Active specification development is now at Change and Configuration Management TC (http://open-services.net/workgroups/ccm-tc)

Want to contribute?

1. Register (/forums/member/register/)
2. Complete Members Agreement (/legal-agreements/members-agreement/)
3. Complete WPA (http://open-services.net/legal-agreements/asset-management-wpa)

Mailing list (http://open-services.net/mailman/listinfo/oslc-assetmgmt_open-services.net)

Workgroup information (http://open-services.net/workgroups/asset-management)

History (http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-Specification/history)   View (http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-Specification)   Links to this page (http://open-services.net/wiki/asset-management/Special:Associated_Pages/OSLC-Asset-Management-2.0-Specification) 2012 September 25 | 09:52 am

Status: 2.0 Specification - September 25th, 2012

This Version
OSLC Asset Management 2.0 Specification (http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-Specification)

Latest Version
http://open-services.net/bin/view/Main/AssetMgSpecificationV2 (Http://Open-Services.Net/Bin/View/Main/AssetMgSpecificationV2.html)

Previous Version
http://open-services.net/bin/view/Main/AssetMgSpecificationV1 (Http://Open-Services.Net/Bin/View/Main/AssetMgSpecificationV1.html)

Authors
SheehanAnderson (http://open-services.net/bin/view/Main/SheehanAnderson)

Contributors
See Contributors section below

Table of Contents

Contents

License

Notation and Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC2119 (http://www.ietf.org/rfc/rfc2119.txt). Domain name examples use RFC2606 (http://tools.ietf.org/html/rfc2606).

Introduction

Asset management systems allow enterprises to catalog, govern, manage, search for, and maintain assets. An asset is anything tangible or intangible that provides value through reference or reuse across a wide audience over an extended period of time. Assets typically have a lifecycle and require a formal process to govern both modification and access. Cataloging assets with standardized taxonomies controls usage and discovery by end users. Assets may also have relationships to and dependencies on other assets.

The term asset in this specification generally refers to either software, documentation, or representations of equipment. While other definitions of an asset may be appropriate, financial instruments are not considered an asset in this specification.

Example software assets may include but are not limited to the final binary output from a software development process, libraries such as a Java archive (JAR) or dynamic-link library (DLL), and installable packages that are distributed through digital distribution platforms such as a repository or app store. Documentation assets may include presentation materials, reports, specifications, blue prints, and instructions. Assets may also represent any type of equipment or structure such as computer hardware, automobiles, pumps, buildings, etc.

Assets are described by a set of properties and zero or more artifacts. An artifact is a file of any type and a set of properties that describe the file.

This specification builds on the Open Services for Lifecycle Collaboration (OSLC) Core v2.0 Specification to define the resources, properties and operations supported by an OSLC Asset Management (OSLC-Asset) provider. Asset Management resources include Assets, Artifacts and supporting resources defined in the OSLC Core specification. The properties defined describe these resources and the relationships between resources. Operations are defined in terms of HTTP methods and MIME type handling. The resources, properties and operations defined do not form a comprehensive interface to Asset Management, but instead target specific integration use cases documented by the OSLC-Asset workgroup.

This specification also defines how Assets and Artifacts are represented in OSLC Services. The Asset Resource (http://open-services.net/bin/view/Main/AssetMgResourceDefinitionsV1) is a set of properties for an Asset, and includes properties that describe Artifacts of an Asset.
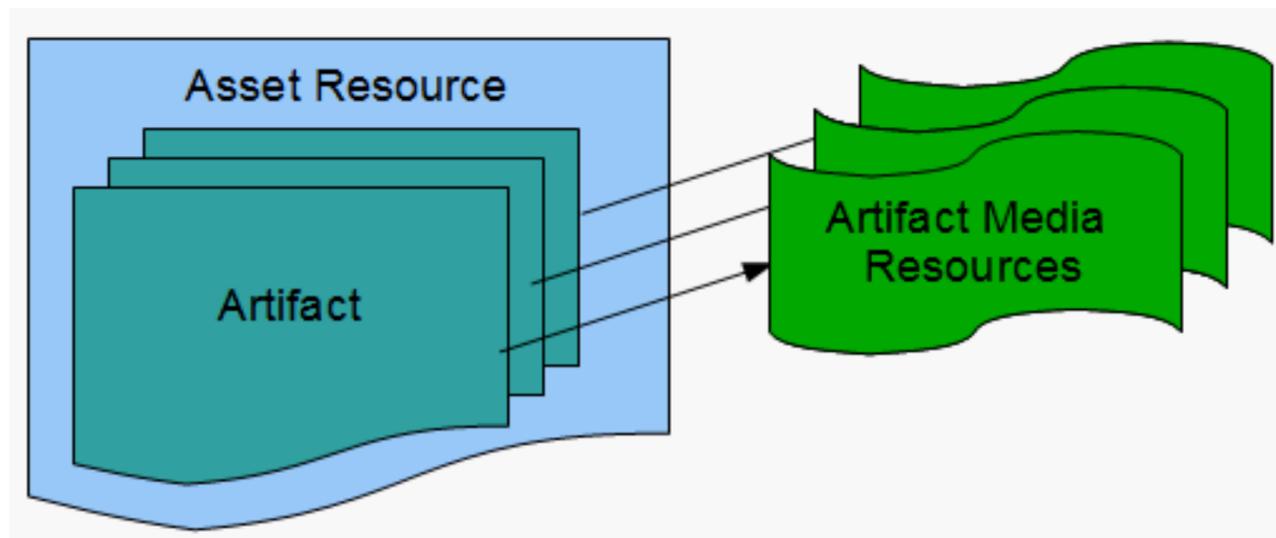
This version of the OSLC Asset specification is demonstrated in the version 2.0 samples (http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-Samples).

Terminology

Asset Consumer - A tool or user that performs asset searching and retrieval activities outlined in the search and retrieve scenarios.

Asset Resource - An Asset Resource has a set of properties representing an asset. These properties include such things as name, description, classification, and the Artifact properties for an Asset.

Asset Query Resource - When filtering using simple query, the representation of a list of Asset Resources.

Asset Submitter - A tool or user that performs asset preparation activities outlined in the publish scenario.

Artifact - This term refers to the properties that describe an Asset Artifact and potentially the Artifact Media referenced by this Artifact. An Asset may have zero or more Artifacts.

Artifact Fragment - Fragment within an Asset Resource that describes an Artifact Media Resource. The term fragment is associated with an Artifact instead of resource because Artifacts are referenced by the Asset's URL with a fragment identifier. Artifacts are considered to be contained within the asset and when the asset is deleted the artifact is also deleted. While two different Artifacts in separate Assets may reference to the same Artifact Media Resource, two different Assets may not contain the same Artifact.

Artifact Media Resource - The Artifact Media Resource represents the actual artifact content. The content can be a presentation, a test case, a binary software component, or any other kind of file.

Property - An attribute of a resource. Typically a property is a name/value pair that describes a certain aspect of the resource. For example, an asset has many properties such as the asset title, subject/short description, version.

Service Provider - An implementation of the OSLC Asset Management specification as a server. OSLC Asset Management clients consume these services.

Work product - This term is often used interchangeably with 'artifact' or 'file'. An asset may have zero or more work products.

Base Requirements

Compliance

This specification is based on OSLC Core Specification (OslcCoreSpecification). OSLC Asset Management 2.0 consumers and service providers **MUST** be compliant with both the core specification and this Asset Management specification, and SHOULD follow all the guidelines and recommendations in both these specifications.

The following table summarizes the requirements from OSLC Core Specification as well as some additional specific to Asset Management. Note that this specification further restricts some of the requirements for OSLC Core Specification. See further sections in this specification or the OSLC Core Specification to get further details on each of these requirements.

Requirement	Level	Meaning
Unknown properties and content	MAY / MUST	OSLC services MAY ignore unknown content and OSLC clients MUST preserve unknown content
Resource Operations	MUST	OSLC service MUST support resource operations via standard HTTP operations
Resource Paging	MAY	OSLC services MAY provide paging for resources but only when specifically requested by client
Partial Resource Representations	MAY / MUST	OSLC services MUST support request for a subset of a resource's properties via the oslc.properties URL parameter retrieval via HTTP GET and MAY support via HTTP PUT
Partial Update	MAY	OSLC services MAY support partial update of resources using patch semantics
Service Provider Resources	MAY / MUST	OSLC service providers MAY provide a Service Provider Catalog and MUST provide a Service Provider resource
Creation Factories	MUST	OSLC service providers MUST provide creation factories to enable resource creation via HTTP POST
Query Capabilities	MUST	OSLC service providers MUST provide query capabilities to enable clients to query for resources
Query Syntax	MUST	OSLC query capabilities MUST support the OSLC Core Query Syntax and MAY use other query syntax
Delegated UI Dialogs	MUST	OSLC Services MUST offer delegated UI dialogs (creation and selections) specified via service provider resource
UI Preview	SHOULD	OSLC Services SHOULD offer UI previews for resources that may be referenced by other resources
HTTP Basic Authentication	MAY	OSLC Services MAY support Basic Auth and should do so only over HTTPS
OAuth Authentication	SHOULD	OSLC Services SHOULD support OAuth and can indicate the required OAuth URLs via the service provider resource
Error Responses	MAY	OSLC Services MAY provide error responses using Core defined error formats
RDF/XML Representations	MUST	OSLC services MUST provide an RDF/XML representation for HTTP GET requests and MUST support RDF/XML representations on POST and PUT requests.
XML Representations	MAY	OSLC services MAY provide a XML representation for HTTP GET, POST and PUT requests that conform to the Core Guidelines for XML.

JSON Representations	SHOULD	OSLC services SHOULD provide JSON representations for HTTP GET, POST and PUT requests that conform to the Core Guidelines for JSON
HTML Representations	SHOULD	OSLC services SHOULD provide HTML representations for HTTP GET requests

Specification Versioning

See OSLC Core Specification Versioning section (http://open-services.net/bin/view/Main/OslcCoreSpecification#Specification_Versioning).

Service providers that support the resource formats and services in this specification **MUST** use HTTP response header of `OSLC-Core-Version` with a value of `2.0` and `OSLC-Asset-Version` with a value of `2.0`. Consumers **MAY** request formats and services defined in this document by providing a HTTP request header of `OSLC-Core-Version` with a value of `2.0`. See section below on Version Compatibility with OSLC Asset 1.0 Specifications.

Namespaces

In addition to the namespace URIs and namespace prefixes `oslc`, `rdf`, `dcterms` and `foaf` defined in the OSLC Core specification (OslcCoreSpecification), OSLC Asset Management defines the namespace URI of http://open-services.net/ns/asset# (http://open-services.net/ns/asset#) with a namespace prefix of `oslc_asset`

Resource Formats

In addition to the requirements for OSLC Core Resource Formats section (http://open-services.net/bin/view/Main/OslcCoreSpecification#Resource_Formats), this section outlines further refinements and restrictions.

For HTTP GET requests on all OSLC Asset Management and OSLC Core defined resource types,

- Asset Management Providers **MUST** provide RDF/XML representations, **SHOULD** provide JSON representations, and **MAY** provide XML representations. The XML and JSON representations **SHOULD** follow the guidelines outlined in the OSLC Core Representations Guidance (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations).
- Asset Management Consumers requesting RDF/XML **SHOULD** be prepared for any valid RDF/XML document. Asset Management Consumers requesting XML or JSON **SHOULD** be prepared for representations that follow the guidelines outlined in the OSLC Core Representations Guidance (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations).
- Asset Management Providers **SHOULD** support an [X]HTML representation and a user interface (UI) preview as defined by UI Preview Guidance (http://open-services.net/bin/view/Main/OslcCoreUiPreview)

For HTTP PUT/POST request formats for resource type of Asset:

- Asset Management Providers **MUST** accept RDF/XML representations, **SHOULD** accept JSON representations, and **MAY** accept XML representations. Asset Management Providers accepting RDF/XML **SHOULD** be prepared for any valid RDF/XML document. For XML or JSON, Asset Management Providers **SHOULD** be prepared for representations that follow the guidelines outlined in the OSLC Core Representations Guidance (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations).

For HTTP GET response formats for Query requests,

Asset Management Providers **MUST** provide RDF/XML representations, **SHOULD** provide JSON and Atom Syndication Format XML representations, and **MAY** provide XML representations.

When Asset Management Consumers request:

- `application/rdf+xml` Asset Management Providers **MUST** respond with RDF/XML representation without restrictions.
- `application/json` Asset Management Providers **SHOULD** respond with JSON representation as defined in the OSLC Core Representations Guidance (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations).
- `application/xml` Asset Management Provider **MAY** respond with OSLC-defined abbreviated XML representation as defined in the OSLC Core Representations Guidance (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations)
- `application/atom+xml` Asset Management Provider **SHOULD** respond with Atom Syndication Format XML representation as defined in the OSLC Core Representations Guidance (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations)
- The Atom Syndication Format XML representation **SHOULD** use RDF/XML representation without restrictions for the atom:content entries representing the resource representations.

See Query Capabilities for additional information when Resource Shapes affect representation.

Content Negotiation

OSLC Core Guidance clearly points to RDF representations (and specifically RDF/XML) as a convention that all OSLC Provider implementations minimally provide and accept. OSLC Asset Provider implementations are strongly encouraged to adopt this convention. Future versions of this specification are expected to require RDF representations for all operations and relax requirements for specialized XML representations.

XML Representation - identified by the `application/xml` content type. Format representation rules are outlined in Core OSLC Core Resource Formats section (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations)

RDF/XML Representation - identified by the `application/rdf+xml` content type. No additional guidance is given. The OSLC Core describes an algorithm for generating consistent formats that are used as examples only.

JSON Representation - identified by the `application/json` content type. Format representation rules are outlined in Core OSLC Core Resource Formats section (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations).

Atom Syndication Format XML Representation - identified by the `application/atom+xml` content type. Format representation rules are outlined in Core OSLC Core Resource Formats section (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations).

Authentication

See OSLC Core Authentication section (http://open-services.net/bin/view/Main/OslcCoreSpecification#Authentication). Asset Management puts no additional constraints on authentication.

Error Responses

See OSLC Core Error Responses section (http://open-services.net/bin/view/Main/OslcCoreSpecification#Error_Responses). Asset Management puts no additional constraints on error responses.

Pagination

OSLC Asset service providers **SHOULD** support pagination of query results and **MAY** support pagination of a single resource's properties as defined by the OSLC Core Specification.

Requesting and Updating Properties

Requesting a Subset of Properties

A client **MAY** request a subset of a resource's properties as well as properties from a referenced resource. In order to support this behavior a service provider **MUST** support the `oslc.properties` and `oslc.prefix` URL parameter on a HTTP GET request on individual resource request or a collection of resources by query. If the `oslc.properties` parameter is omitted on the request, then all resource properties **MUST** be provided in the response.

Updating a Subset of Properties

A client **MAY** request that a subset of a resource's properties be updated by identifying those properties to be modified using the `oslc.properties` URL parameter on a HTTP PUT request.

If the parameter `oslc.properties` contains a valid resource property on the request that is not provided in the content, the server **MUST** set the resource's property to a null or empty value. If the parameter `oslc.properties` contains an invalid resource property, then a `409 Conflict` **MUST** be returned.

Labels for Relationships

Asset relationships to other resources are represented as properties whose values are the URI of the object or target resource. When an Asset relationship property is to be presented in a user interface, it may be helpful to provide an informative and useful textual label for that relationship instance. (This in addition to the relationship property URI and the object resource URI, which are also candidates for presentation to a user.) To this end, OSLC providers **MAY** support a `dcterms:title` link property in Asset resource representations, using the anchor approach outlined in the OSLC Core Links Guidance. In addition to the textual label, the Asset resource definition defines several properties that can be used to describe the relationship from one Asset resource to another Asset resource.

RDF/XML and XML example using reified statement:

```
<rdf:RDF
    xmlns:dcterms="http://purl.org/dc/terms/"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:oslc_asset="http://open-services.net/ns/asset#">

    <oslc_asset:Asset rdf:about="http://example.com/assets/CDFE4153-9271-5CA6-425A-4CA6BE7BD7CA/1.0">
        <dcterms:relation rdf:ID="relationship1" rdf:resource="http://example.com/assets/52C4702C-7CA2-E46E-D373-620482ECBEED/1.0"/>
    </oslc_asset:Asset>

    <rdf:Description rdf:about="#relationship1">
        <dcterms:title>Build Configuration</dcterms:title>
    </rdf:Description>
</rdf:RDF>
```

Custom Properties

Asset management providers **SHOULD** allow consumers to define custom properties. If providers allow consumers to define custom properties, they **MUST** allow the consumer to optionally specify an HTTP URI for that data element, and the specified URI **MUST** be used in the RDF representation of all resources that contain the data element. The specified URI **MAY** be part of an industry standard such as DC, FOAF, or OSLC, or it **MAY** be defined by the user's organization. If the URI is defined by the user's organization, then they SHOULD make it dereferencable as per the W3C Best Practices for Publishing RDF Vocabularies (http://www.w3.org/TR/swbp-vocab-pub/).

If the user does not specify the optional URI, then the provider **MUST** use a default generated HTTP URI. The generated URI **SHOULD** be as human-readable as possible. One method for achieving human-readability is to apply a human-friendly name mangling algorithm to the user-defined label for the data element. The tool **SHOULD** make the generated URI dereferencable as per the W3C Best Practices for Publishing RDF Vocabularies (http://www.w3.org/TR/swbp-vocab-pub/). The published RDF vocabulary **SHOULD** include any relevant descriptive information provided the user as part of the data element definition, e.g. the label and description.

OSLC Asset Management Resource Definitions

Asset

An Asset is anything tangible or intangible that is capable of being owned or controlled to produce value, and is held to have positive economic value. This includes not only software products, but also buildings, servers, automobiles, pumps, oil wells, etc. These properties include such things as name, description, classification, and the Artifact properties of an Asset.

The Asset resource properties are not limited to the ones defined in this specification, service providers may provide additional properties. It is recommended that any additional properties exist in their own unique namespace and not use the namespaces defined in these specifications.

- **Name:** Asset
- **Type URI** http://open-services.net/ns/asset#Asset (http://open-services.net/ns/asset#Asset)

Prefixed Name	Occurs	Read-only	Value-type	Represen-tation	Range	Description
OSLC Core: Common Properties						
dcterms:identifier	zero-or-one	True	String	n/a	n/a	A unique identifier for a resource. Assigned by the service provider when a resource is created.
dcterms:title	exactly-one	unspecified	XMLLiteral	n/a	n/a	Short name identifying a resource, often used as an abbreviated identifier for presentation to end-users. SHOULD include only content that is valid inside an XHTML element.
dcterms:description	zero-or-one	unspecified	XMLLiteral	n/a	n/a	Descriptive text (reference: Dublin Core) about resource represented as rich text in XHTML content. SHOULD include only content that is valid and suitable inside an XHTML element.
dcterms:creator	zero-or-many	unspecified	Either Resource or Local Resource	Either Reference or Inline	any	Creator or creators of resource (reference: Dublin Core). It is likely that the target resource will be a foaf:Person (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixA#foaf_Person_Resource) but that is not necessarily the case
dcterms:contributor	zero-or-many	unspecified	Either Resource or Local Resource	Either Reference or Inline	any	The person(s) who are responsible for this asset. (reference: Dublin Core). It is likely that the target resource will be a foaf:Person (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixA#foaf_Person_Resource) but that is not necessarily the case.
dcterms:created	zero-or-one	True	DateTime	n/a	n/a	Timestamp of resource creation (reference: Dublin Core).
dcterms:modified	zero-or-one	True	DateTime	n/a	n/a	Timestamp latest resource modification (reference: Dublin Core).
rdf:type	zero-or-many	unspecified	Resource	Reference	n/a	The resource type URIs. One reference should have the value of http://open-services.net/ns/asset#Asset (http://open-services.net/ns/asset#Asset).
oslc:serviceProvider	zero-or-many	unspecified	Resource	Reference	oslc:ServiceProvider (http://open-services.net/bin/view/Main/OslcCoreSpecification#Service_Provider_Resources)	The scope of a resource is a URI for the resource's OSLC Service Provider.
oslc:instanceShape	zero-or-one	unspecified	Resource	Reference	oslc:ResourceShape (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixA#oslc_ResourceShape_Resource)	Resource Shape that provides hints as to resource property value-types and allowed values.
Prefixed Name	**Occurs**	**Read-only**	**Value-type**	**Representation**	**Range**	**Description**
OSLC Asset: Start of additional properties						
oslc_asset:guid	zero-or-one	unspecified	String	n/a	n/a	An identifier for the asset. Assigned by the service provider when a resource is created. Different versions of the same asset will share the same identifier.
oslc_asset:version	zero-or-one	unspecified	String	n/a	n/a	The version of the asset. Possible values may include '1.0', '2.0', etc.
dcterms:abstract	zero-or-one	unspecified	XMLLiteral	n/a	n/a	Short description (reference: Dublin Core) or often a single line summary of the resource
dcterms:type	zero-or-one	unspecified	Resource	Reference	n/a	The type of the asset based on values defined by the service provider. This specification does not define the resource for this property, however it should contain a dcterms:title property.
oslc_asset:state	zero-or-one	unspecified	Either Resource or Local Resource	Either Reference or Inline	any	Used to indicate the state of the asset based on values defined by the service provider. This specification does not define the resource for this property, however it should contain a dcterms:title property.
oslc_asset:categorization	zero-or-many	unspecified	Either Resource or Local Resource	Either Reference or Inline	any	A categorization to classify an asset. The category schema values are defined by the service provider. This specification does not define the resource for this property, however it should contain a dcterms:title property.
oslc_asset:manufacturer	zero-or-one	unspecified	String	n/a	n/a	The name of the asset manufacturer.
oslc_asset:model	zero-or-one	unspecified	String	n/a	n/a	The value of the asset model.
oslc_asset:serialNumber	zero-or-one	unspecified	String	n/a	n/a	The serial number assigned by the asset manufacturer.
oslc_asset:tag	zero-or-many	unspecified	String	n/a	n/a	Specifies the asset tag value for an Asset. Asset tags are typically human readable labels. For hardware assets, these tags are durable, securely attached to equipment, and may also be readable by barcode and/or RFID.
oslc_asset:artifact	zero-or-many	unspecified	Local Resource	Inline	oslc_asset:Artifact	An Artifact fragment contained in this Asset resource.

Prefixed Name	Occurs	Read-only	Value-type	Representation	Range	Description
oslc_asset:artifactFactory	exactly-one	unspecified	Resource	Reference	any	Resource URI used to post new artifacts to the asset.
Relationship Properties: This grouping of properties are used to identify relationships between resources managed by the current or other OSLC Service Providers						
dcterms:relation	zero-or-many	False	Resource	Reference	any	This relationship is loosely coupled and has no specific meaning. Details about this relationship may be included in a reified statement.

Prefixed Name	Occurs	Read-only	Value-type	Representation	Range	Description
Related Resource Reified Statement Properties: This grouping of properties is used to provide additional details about a relationship in a reified statement						
oslc_asset:relationshipType	zero-or-one	unspecified	Either Resource or Local Resource	Either Reference or Inline	any	The type of this relationship from the perspective of the dcterms:relation resource based on values defined by the service provider. This specification does not define the resource for this property, however it should contain a dcterms:title property.
dcterms:creator	zero-or-many	unspecified	Either Resource or Local Resource	Either Reference or Inline	any	Creator or creators of the relationship (reference: Dublin Core). It is likely that the target resource will be a **foaf:Person** (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixA#foaf_Person_Resource) but that is not necessarily the case.
dcterms:created	zero-or-one	True	DateTime	n/a	n/a	Timestamp of the relationship creation (reference: Dublin Core).
dcterms:modified	zero-or-one	True	DateTime	n/a	n/a	Timestamp of the latest relationship modification (reference: Dublin Core).
oslc_asset:state	zero-or-one	unspecified	Either Resource or Local Resource	Either Reference or Inline	any	Used to indicate the state of the relationship based on values defined by the service provider. This specification does not define the resource for this property, however it should contain a dcterms:title property.

Artifact

An Artifact is a fragment within an asset resource that describes an Artifact Media Resource. An artifact fragment may only be updated though an Asset resource and it is not possible to perform operations directly on the Artifact fragment

The Artifact fragment properties are not limited to the ones defined in this specification, service providers may provide additional properties. It is recommended that any additional properties exist in their own unique namespace and not use the namespaces defined in these specifications.

- **Name:** `Artifact`
- **Type URI** http://open-services.net/ns/asset#Artifact (http://open-services.net/ns/asset#Artifact)

Prefixed Name	Occurs	Read-only	Value-type	Represen-tation	Range	Description
OSLC Artifact Fragment: The grouping of properties that define an artifact						
rdf:type	zero-or-many	unspecified	Resource	Reference	n/a	The resource type URIs. One reference should have the value of http://open-services.net/ns/asset#Artifact. (http://open-services.net/ns/asset#Artifact.)
dcterms:title	exactly-one	unspecified	XMLLiteral	n/a	n/a	The name of the artifact. Title (reference: Dublin Core) of the resource represented in text XHTML content. SHOULD include only content that is valid inside an XHTML element.
oslc:label	zero-or-one	unspecified	XMLLiteral	n/a	n/a	The label of the artifact. Subject (reference: Dublin Core) an abbreviated presentation for the end-users. SHOULD include only content that is valid inside an XHTML element.
oslc_asset:content	exactly-one	unspecified	Resource	Reference	**Artifact Media**	The media resource reference URI (the artifact bytes).
dcterms:format	zero-or-one	unspecified	String	n/a	n/a	The mime type of the artifact media resource.
oslc_asset:size	zero-or-one	True	Integer	n/a	n/a	The size of the artifact media resource in bytes.
dcterms:created	zero-or-one	True	DateTime	n/a	n/a	Timestamp of resource creation (reference: Dublin Core).
dcterms:modified	zero-or-one	True	DateTime	n/a	n/a	Timestamp last latest resource modification (reference: Dublin Core).

Artifact Media

The artifact media represents the actual content of an artifact. The URI for the artifact media resource is defined by the `oslc_asset:content` property of an artifact fragment within an asset resource.

A client may publish a new artifact to an asset making a POST request to the asset's artifact factory URI as defined by the `oslc_asset:artifactFactory` property of the asset resource. The request **MUST** contain an oslc_asset.name header indicating the path and name of the new artifact and the body of the POST **MUST** contain the contents of the artifact. A successful response from the provider **MUST** return a status code of 201 and a Location Header indicating the URI of the Media Resource. In addition, the provider **MUST** add a new artifact fragment to the asset and fill in the mandatory properties including size, and creation time. The client may later update the non-read only properties of this fragment.

See the Publish an artifact (/wiki/asset-management/OSLC-Asset-Management-2.0-Samples/#Publish-an-artifact) sample for additional details.

Providers **MUST** also support URI artifacts. URI artifacts do not contain content, but instead the `oslc_asset:content` property's value is a URI external to the asset. To create a URI artifact, clients POST an artifact resource to the artifact factory URI. The request **MUST** contain an `oslc_asset.name` header indicating the path and name of the new artifact URI. A successful response from the provider **MUST** return a status code of 201 and a Location Header indicating the URI of the artifact URI. In addition, the provider MUST add a new artifact fragment to the asset and fill in the mandatory properties such as creation time. The client may later update the non-read only properties of this fragment.

See the Publish a URI artifact (/wiki/asset-management/OSLC-Asset-Management-2.0-Samples/#Publish-a-URI-artifact) sample for additional details.

Asset Management Service Provider Capabilities

Service Discovery and Description

OSLC Asset Management service providers **MUST** provide a Service Provider Resource (http://open-services.net/bin/view/Main/OslcCoreSpecification#Service_Provider_Resources) that can be retrieved at a implementation dependent URI.

OSLC Asset Management service providers **MAY** provide a Service Provider Catalog Resource (http://open-services.net/bin/view/Main/OslcCoreSpecification#Service_Provider_Catalog_Resources) that can be retrieved at a implementation dependent URI.

OSLC Asset Management service providers **MUST** provide a `oslc:serviceProvider` property for their defined resources that will be the URI to a Service Provider Resource (http://open-services.net/bin/view/Main/OslcCoreSpecification#Service_Provider_Resources).

OSLC Asset Management service providers **MUST** supply a value of http://open-services.net/ns/asset# (http://open-services.net/ns/asset#) for the property `oslc:domain` on either `oslc:Service` or `oslc:ServiceProviderCatalog` resources.

Creation Factories

OSLC Asset Management service providers **MUST** support [Creation Factories (http://open-services.net/bin/view/Main/OslcCoreSpecification#Creation_Factories)](http://open-services.net/bin/view/Main/OslcCoreSpecification#Creation_Factories) and list them in the Service Provider Resource as defined by OSLC Core. OSLC Asset Management service providers **SHOULD** support [Resource Shapes for Creation Factories (http://open-services.net/bin/view/Main/OslcCoreSpecification#Resource_Shapes)](http://open-services.net/bin/view/Main/OslcCoreSpecification#Resource_Shapes) as defined in [OSLC Core Specification (http://open-services.net/bin/view/Main/OslcCoreSpecification#Resource_Shapes)](http://open-services.net/bin/view/Main/OslcCoreSpecification#Resource_Shapes)

Query Capabilities

OSLC Asset Management service providers **MUST** support the [Query Capabilities (http://open-services.net/bin/view/Main/OslcCoreSpecification#Query_Capabilities)](http://open-services.net/bin/view/Main/OslcCoreSpecification#Query_Capabilities) as defined by OSLC Core. OSLC Asset Management service providers **SHOULD** support [Resource Shapes for Query Capability (http://open-services.net/bin/view/Main/OslcCoreSpecification#Resource_Shapes)](http://open-services.net/bin/view/Main/OslcCoreSpecification#Resource_Shapes) as defined in [OSLC Core Specification (http://open-services.net/bin/view/Main/OslcCoreSpecification#Resource_Shapes)](http://open-services.net/bin/view/Main/OslcCoreSpecification#Resource_Shapes)

The Query Capability **MUST** support these parameters:

- `oslc.where`
- `oslc.select`
- `oslc.properties`
- `oslc.prefix`

If shape information is NOT present with the Query Capability, service providers **SHOULD** use these default properties to contain the result:

- For RDF/XML and XML, use `rdf:Description` and `rdfs:member` as defined in [OSLC Core RDF/XML Examples (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations#Specifying_the_shape_of_a_query)](http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations#Specifying_the_shape_of_a_query)
- For JSON, the query results are contained within `oslc:results` array. See [OSLC Core Representation Guidance for JSON (http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations#Guidelines_for_JSON)](http://open-services.net/bin/view/Main/OSLCCoreSpecAppendixRepresentations#Guidelines_for_JSON)

Delegated User Interface Dialogs

OSLC Asset Management service providers **SHOULD** support the selection of resources by delegated web-based user interface dialogs [Delegated UIs (http://open-services.net/bin/view/Main/OslcCoreSpecification#Delegated_User_Interface_Dialogs)](http://open-services.net/bin/view/Main/OslcCoreSpecification#Delegated_User_Interface_Dialogs) as defined by OSLC Core.

OSLC Asset Management service providers **MAY** support the creation of resources by delegated web-based user interface dialogs [Delegated UIs (http://open-services.net/bin/view/Main/OslcCoreSpecification#Delegated_User_Interface_Dialogs)](http://open-services.net/bin/view/Main/OslcCoreSpecification#Delegated_User_Interface_Dialogs) as defined by OSLC Core.

OSLC Asset Management service providers **MAY** support the pre-filling of creation dialogs based on the definition at [Delegated UIs (http://open-services.net/bin/view/Main/OslcCoreSpecification#Delegated_User_Interface_Dialogs)](http://open-services.net/bin/view/Main/OslcCoreSpecification#Delegated_User_Interface_Dialogs).

The service providers supports the delegated UIs as follows:

Asset Management Resource	Selection	Creation
Asset	SHOULD	MAY

Version Compatibility with 1.0 Specifications

The goal is to provide a smooth transition to 2.0 for both Consumers and Providers. This section will clarify the usage of 1.0 media types so that Providers can support both 1.0 and 2.0 Consumers when HTTP requests are made for a resource with the same URI.

Media Types

For an Asset Resource format identification of RDF/XML and XML, the media type used for this representation SHOULD be `application/rdf+xml` or `application/xml`.

For an Asset Resource format identification of JSON, the media type used for this representation SHOULD be `application/json`.

Appendix A: Samples

(this section is informative)

See [OSLC Asset Management 2.0 Samples (http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-Samples)](http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-Samples).

Appendix B: Resource Shapes

(this section is informative)

See [OSLC Asset Management 2.0 Appendix B: Resource Shapes (http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-Appendix-B%3A-Resource-Shapes)](http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-Appendix-B%3A-Resource-Shapes)

Appendix C: Notices and References

Contributors

- Brad Sandler (IBM)
- Dave Johnson (IBM)
- Edwin Freekenhorst (Shell)
- Eric Bordeau (IBM)
- Gili Mendel (IBM; OSLC Asset Management Lead)
- Grant Larsen (IBM)
- Hari Padmanabhan (IBM)
- Jaimin Patel (WebLayers)
- John Favazza (WebLayers)
- Kevin Bauer (IBM)
- Randy Lexvold (the Emphasys Group)
- Ren Renganathan (Citigroup)
- Scott Bosworth (IBM)
- Sheehan Anderson (IBM)
- Srimanth Gunturi (IBM)

Reporting Issues on the Specification

The working group participants who author and maintain this working draft specification, monitor a distribution list where issues or questions can be raised, see [Asset Management Mailing List (http://open-services.net/mailman/listinfo/oslc-assetmgmt_open-services.net)](http://open-services.net/mailman/listinfo/oslc-assetmgmt_open-services.net).

Also the issues found with this specification and their resolution can be found at [OSLC Asset Management 2.0 specification issues (http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-specification-issues)](http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-specification-issues).

Intellectual Property Covenant

The members of the Working Group (or as appropriate, their employers) have documented a Patent Non-Assertion Covenant for implementations of the Asset Management 2.0 Specification, as described in the open-services.net [Terms of Use (/terms/)](/terms/). Details of the Covenant may be found [here (http://open-services.net/bin/view/Main/AssetMgSpecificationV2Covenant)](http://open-services.net/bin/view/Main/AssetMgSpecificationV2Covenant).

References

- OSLC-Asset 1.0 - [OSLC Asset Management Specifications1.0 (http://open-services.net/bin/view/Main/AssetMgSpecificationV1)](http://open-services.net/bin/view/Main/AssetMgSpecificationV1)
- OSLC Core - [OSLC Core Specification 2.0 (http://open-services.net/bin/view/Main/OslcCoreSpecification)](http://open-services.net/bin/view/Main/OslcCoreSpecification)
- ATOM - [RFC4287 - Atom Syndication Format (http://tools.ietf.org/html/rfc4287)](http://tools.ietf.org/html/rfc4287)
- Dublin Core 1.1 - [Dublin Core Metadata Element Set, Version 1.1 (http://dublincore.org/documents/2010/10/11/dces/)](http://dublincore.org/documents/2010/10/11/dces/)
- FOAF - [Friend of a Friend (FOAF) v0.98 (http://xmlns.com/foaf/spec/20100809.html)](http://xmlns.com/foaf/spec/20100809.html)
- HTTP 1.1 - [Hyper-text Transfer Protocol (HTTP/1.1) (http://tools.ietf.org/html/rfc2616)](http://tools.ietf.org/html/rfc2616)
- JSON - [JavaScript Object Notation (http://json.org/)](http://json.org/)
- OAuth 1.0a - [RFC5849 - The OAuth 1.0 Protocol (http://tools.ietf.org/html/rfc5849)](http://tools.ietf.org/html/rfc5849)
- RDF/XML Concepts - [RDF/XML Concepts and Abstract Syntax (http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/)](http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/)
- RDF/XML Syntax - [RDF / XML Syntax Specification (Revised) (http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/)](http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/)
- URI Syntax - [URI Generic Syntax (http://tools.ietf.org/html/rfc3986)](http://tools.ietf.org/html/rfc3986)

- XML Namespaces - <u>Namespaces in XML 1.0 (Third Edition) (http://www.w3.org/TR/REC-xml-names/)</u>
- XSD Datatypes - <u>XML Schema Part 2: Datatypes Second Edition (http://www.w3.org/TR/xmlschema-2)</u>