OSLC Asset Management 2.0 Specification

Want to contribute?

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

Want to contribute?

1. Register (/forums/member/register/)
2. Complete Members Agreement (/legal-agreements/members-agreement/)
3. Complete WPA (/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/)


Links to this page (/http://open-services.net/wiki/asset-management/OSLC-Asset-Management-2.0-Specification/)

Status: 2.0 Specification - September 25th, 2012

This Version

Latest Version

Previous Version

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

Contributors
See Contributors section below

Table of Contents

Contents

Introduction
  • Terminology

Base Requirements
  • Compliance
  • Specification Versioning
  • Namespaces
  • Resource Formats
    • Content Negotiation
  • Authentication
  • Error Responses
  • Parsing
  • Requesting and Updating Properties
    • Requesting a Subset of Properties
    • Updating a Subset of Properties
  • Labels for Relationships
  • Custom Properties

OSLC Asset Management Resource Definitions
  • Asset
  • Artifact
  • Artifact Media

Asset Management Service Provider Capabilities
  • Service Discovery and Description
  • Creation Factories
  • Query Capabilities
  • Delegated User Interface Dialogs

Version Compatibility with 1.0 Specifications
  • Media Types

Appendix A: Samples
Appendix B: Resource Shapes
Appendix C: Notices and References
  • Contributors
  • Reporting Issues on the Specification
  • Intellectual Property Covenant
  • References

License
Notation and Conventions


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, blueprints, 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/AssetResourceDefinitionsV1) is a set of properties for an Asset, and includes properties that describe Artifacts of an Asset.

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.

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

link property in Asset resource representations, using the anchor approach outlined in the OSLC Core Links Guidance. In ... the textual label, the Asset resource definition defines several properties that can be used to describe the relationship

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

Asset relationships to other resources are represented as properties whose values are the URI of the object or target resource. ... an Asset relationship property is to be presented in a user interface, it may be helpful to provide an informative and

Labels for Relationships

Invalid resource property, then a

If the parameter

A client

Updating a Subset of Properties

A client

Requesting and Updating Properties

OSLC Asset service providers

Pagination

See

Error Responses

See

Authentication

services.net/bin/view/Main/OslcCoreSpecification#Error_Responses

services.net/bin/view/Main/OslcCoreSpecification#Authentication

OSLC Core Error Responses section

see section below on Version Compatibility with OSLC Asset 1.0 Specifications.

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

OSLC services SHOULD provide XML representations for HTTP GET requests

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 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 parameters 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 determine: 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:
### 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.

<table>
<thead>
<tr>
<th>Prefixed Name</th>
<th>Occurs</th>
<th>Read-only</th>
<th>Value-type</th>
<th>Representation</th>
<th>Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>oslc:guid</td>
<td>zero-or-one</td>
<td>True</td>
<td>String</td>
<td>n/a</td>
<td>n/a</td>
<td>A unique identifier for a resource. Assigned by the service provider when a resource is created.</td>
</tr>
<tr>
<td>oslc:asset.version</td>
<td>zero-or-one</td>
<td>True</td>
<td>String</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>oslc:asset.description</td>
<td>zero-or-one</td>
<td>True</td>
<td>XMLLiteral</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>oslc:asset:manufacturer</td>
<td>zero-or-many</td>
<td>false</td>
<td>Resource or Local Resource</td>
<td>Either Reference or Inline</td>
<td>any</td>
<td>Creator or creators of resource (reference: Dublin Core). It is likely that the target resource will be a foaf:Person (<a href="http://open-services.net/bin/view/Main/OslcCoreSpecAppendixA#foaf_Person_Resource">http://open-services.net/bin/view/Main/OslcCoreSpecAppendixA#foaf_Person_Resource</a>) but that is not necessarily the case.</td>
</tr>
<tr>
<td>oslc:asset:state</td>
<td>zero-or-one</td>
<td>True</td>
<td>DateTime</td>
<td>n/a</td>
<td>n/a</td>
<td>Timestamp of resource creation (reference: Dublin Core).</td>
</tr>
<tr>
<td>oslc:asset:serialNumber</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>Timestamp latest resource modification (reference: Dublin Core).</td>
</tr>
<tr>
<td>oslc:asset:model</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>The resource type URIs. One resource should have the value of <a href="http://open-services.net/ns/asset#Asset">http://open-services.net/ns/asset#Asset</a> (<a href="http://open-services.net/ns/asset#Asset">http://open-services.net/ns/asset#Asset</a>).</td>
</tr>
<tr>
<td>oslc:asset:path</td>
<td>zero-or-one</td>
<td>False</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>The scope of a resource is a URI for the resource’s OSLC Service Provider.</td>
</tr>
</tbody>
</table>

#### OSLC Core: Common Properties

<table>
<thead>
<tr>
<th>Prefixed Name</th>
<th>Occurs</th>
<th>Read-only</th>
<th>Value-type</th>
<th>Representation</th>
<th>Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dcterms:identifier</td>
<td>zero-or-one</td>
<td>True</td>
<td>String</td>
<td>n/a</td>
<td>n/a</td>
<td>A unique identifier for a resource. Assigned by the service provider when a resource is created.</td>
</tr>
<tr>
<td>dcterms:title</td>
<td>exactly-one</td>
<td>unspecified</td>
<td>XMLLiteral</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>dcterms:description</td>
<td>zero-or-one</td>
<td>True</td>
<td>XMLLiteral</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>dcterms:creator</td>
<td>zero-or-many</td>
<td>False</td>
<td>Resource or Local Resource</td>
<td>Either Reference or Inline</td>
<td>any</td>
<td>Creator or creators of resource (reference: Dublin Core). It is likely that the target resource will be a foaf:Person (<a href="http://open-services.net/bin/view/Main/OslcCoreSpecAppendixA#foaf_Person_Resource">http://open-services.net/bin/view/Main/OslcCoreSpecAppendixA#foaf_Person_Resource</a>) but that is not necessarily the case.</td>
</tr>
<tr>
<td>dcterms:contributor</td>
<td>zero-or-many</td>
<td>False</td>
<td>Resource or Local Resource</td>
<td>Either Reference or Inline</td>
<td>any</td>
<td>The person(s) who are responsible for this asset. (reference: Dublin Core). It is likely that the target resource will be a foaf:Person (<a href="http://open-services.net/bin/view/Main/OslcCoreSpecAppendixA#foaf_Person_Resource">http://open-services.net/bin/view/Main/OslcCoreSpecAppendixA#foaf_Person_Resource</a>) but that is not necessarily the case.</td>
</tr>
<tr>
<td>dcterms:created</td>
<td>zero-or-one</td>
<td>True</td>
<td>DateTime</td>
<td>n/a</td>
<td>n/a</td>
<td>Timestamp of resource creation (reference: Dublin Core).</td>
</tr>
<tr>
<td>dcterms:modified</td>
<td>zero-or-one</td>
<td>True</td>
<td>DateTime</td>
<td>n/a</td>
<td>n/a</td>
<td>Timestamp latest resource modification (reference: Dublin Core).</td>
</tr>
<tr>
<td>rdf:type</td>
<td>zero-or-many</td>
<td>False</td>
<td>Resource</td>
<td>n/a</td>
<td>n/a</td>
<td>The resource type URIs. One resource should have the value of <a href="http://open-services.net/ns/asset#Asset">http://open-services.net/ns/asset#Asset</a> (<a href="http://open-services.net/ns/asset#Asset">http://open-services.net/ns/asset#Asset</a>).</td>
</tr>
<tr>
<td>oslc:serviceProvider</td>
<td>zero-or-many</td>
<td>False</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>The scope of a resource is a URI for the resource’s OSLC Service Provider.</td>
</tr>
<tr>
<td>oslc:instanceShape</td>
<td>zero-or-one</td>
<td>False</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>Resource Shape that provides hints as to resource property value-types and allowed values.</td>
</tr>
</tbody>
</table>

#### Additional properties

<table>
<thead>
<tr>
<th>Prefixed Name</th>
<th>Occurs</th>
<th>Read-only</th>
<th>Value-type</th>
<th>Representation</th>
<th>Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>oslc:asset:artifact</td>
<td>zero-or-many</td>
<td>True</td>
<td>Local</td>
<td>Inline</td>
<td>oslc_asset:Artifact</td>
<td>An artifact indicating 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.</td>
</tr>
<tr>
<td>oslc:asset:tag</td>
<td>zero-or-many</td>
<td>True</td>
<td>String</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>oslc:asset:format</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>oslc:asset:version</td>
<td>zero-or-one</td>
<td>True</td>
<td>String</td>
<td>n/a</td>
<td>n/a</td>
<td>The version of the asset. Possible values may include ‘1.0’, ‘2.0’, etc.</td>
</tr>
<tr>
<td>oslc:asset:state</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>Short description (reference: Dublin Core) or often a single line summary of the resource</td>
</tr>
<tr>
<td>oslc:asset:serialNumber</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>oslc:asset:manufacter</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
</tbody>
</table>

### W3C Best Practices for Publishing RDF Vocabularies

The W3C Best Practices for Publishing RDF Vocabularies guide provides advice for publishing RDF vocabularies that are intended to be human-readable. The values in this table are based on these best practices.

<table>
<thead>
<tr>
<th>Prefixed Name</th>
<th>Occurs</th>
<th>Read-only</th>
<th>Value-type</th>
<th>Representation</th>
<th>Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>oslc:asset:guid</td>
<td>zero-or-one</td>
<td>True</td>
<td>String</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>oslc:asset:version</td>
<td>zero-or-one</td>
<td>True</td>
<td>String</td>
<td>n/a</td>
<td>n/a</td>
<td>The version of the asset. Possible values may include ‘1.0’, ‘2.0’, etc.</td>
</tr>
<tr>
<td>dcterms:abstract</td>
<td>zero-or-one</td>
<td>True</td>
<td>XMLLiteral</td>
<td>n/a</td>
<td>n/a</td>
<td>Short description (reference: Dublin Core) or often a single line summary of the resource</td>
</tr>
<tr>
<td>oslc:asset:state</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>oslc:asset:serialNumber</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>oslc:asset:manufacturer</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>The name of the asset manufacturer.</td>
</tr>
<tr>
<td>oslc:asset:format</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>The value of the asset model.</td>
</tr>
<tr>
<td>oslc:asset:serialNumber</td>
<td>zero-or-one</td>
<td>True</td>
<td>Reference</td>
<td>n/a</td>
<td>n/a</td>
<td>The serial number assigned by the asset manufacturer.</td>
</tr>
<tr>
<td>oslc:asset:tag</td>
<td>zero-or-many</td>
<td>True</td>
<td>String</td>
<td>n/a</td>
<td>n/a</td>
<td>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.</td>
</tr>
<tr>
<td>oslc:asset:artifact</td>
<td>zero-or-many</td>
<td>True</td>
<td>Local</td>
<td>Inline</td>
<td>oslc_asset:Artifact</td>
<td>An artifact indicating 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.</td>
</tr>
</tbody>
</table>
**Creation Factories**

**OSLC Asset Management service providers**

**Service Discovery and Description**

<table>
<thead>
<tr>
<th><strong>Prefixed Name</strong></th>
<th><strong>Occurs</strong></th>
<th><strong>Read-only</strong></th>
<th><strong>Value-type</strong></th>
<th><strong>Representation Range</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>oslc_asset:artifactFactory</td>
<td>exactly-unspecified</td>
<td>Resource</td>
<td>Reference</td>
<td>any</td>
<td>Resource URI used to post new artifacts to the asset.</td>
</tr>
<tr>
<td>dcterms:relation</td>
<td>zero-or-many</td>
<td>Resource</td>
<td>Reference</td>
<td>any</td>
<td>This relationship is loosely coupled and has no specific meaning. Details about this relationship may be included in a reified statement.</td>
</tr>
</tbody>
</table>

**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)

**OSLC Artifact Fragment: The grouping of properties that define an artifact**

<table>
<thead>
<tr>
<th><strong>Prefixed Name</strong></th>
<th><strong>Occurs</strong></th>
<th><strong>Read-only</strong></th>
<th><strong>Value-type</strong></th>
<th><strong>Representation</strong></th>
<th><strong>Range</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>rdf:type</td>
<td>zero-or-many</td>
<td>Resource</td>
<td>Reference</td>
<td>n/a</td>
<td>The resource type URIs. One reference should have the value of <a href="http://open-services.net/ns/asset#Artifact">http://open-services.net/ns/asset#Artifact</a>.</td>
<td></td>
</tr>
<tr>
<td>dcterms:title</td>
<td>exactly-one</td>
<td>unspecified XMLLiteral</td>
<td>n/a</td>
<td>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.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>oslc:label</td>
<td>zero-or-one</td>
<td>unspecified XMLLiteral</td>
<td>n/a</td>
<td>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.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>oslc_asset:content</td>
<td>exactly-one</td>
<td>Resource</td>
<td>Reference</td>
<td>Artifact Media</td>
<td>The media resource reference URI (the artifact bytes).</td>
<td></td>
</tr>
<tr>
<td>dcterms:format</td>
<td>zero-or-one</td>
<td>unspecified String</td>
<td>n/a</td>
<td>The mime type of the artifact media resource.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>oslc_asset:size</td>
<td>zero-or-one</td>
<td>True</td>
<td>Integer</td>
<td>n/a</td>
<td>The size of the artifact media resource in bytes.</td>
<td></td>
</tr>
<tr>
<td>dcterms:created</td>
<td>zero-or-one</td>
<td>True</td>
<td>DateTime</td>
<td>n/a</td>
<td>Timestamp of resource creation (reference: Dublin Core).</td>
<td></td>
</tr>
<tr>
<td>dcterms:modified</td>
<td>zero-or-one</td>
<td>True</td>
<td>DateTime</td>
<td>n/a</td>
<td>Timestamp last resource modification (reference: Dublin Core).</td>
<td></td>
</tr>
</tbody>
</table>

**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:content, 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](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](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](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**

<table>
<thead>
<tr>
<th><strong>Prefixed Name</strong></th>
<th><strong>Occurs</strong></th>
<th><strong>Read-only</strong></th>
<th><strong>Value-type</strong></th>
<th><strong>Representation Range</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>oslc_asset:artifactFactory</td>
<td>exactly-unspecified</td>
<td>Resource</td>
<td>Reference</td>
<td>any</td>
<td>Resource URI used to post new artifacts to the asset.</td>
</tr>
<tr>
<td>dcterms:relation</td>
<td>zero-or-many</td>
<td>Resource</td>
<td>Reference</td>
<td>any</td>
<td>This relationship is loosely coupled and has no specific meaning. Details about this relationship may be included in a reified statement.</td>
</tr>
</tbody>
</table>

**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)

**OSLC Artifact Fragment: The grouping of properties that define an artifact**

<table>
<thead>
<tr>
<th><strong>Prefixed Name</strong></th>
<th><strong>Occurs</strong></th>
<th><strong>Read-only</strong></th>
<th><strong>Value-type</strong></th>
<th><strong>Representation</strong></th>
<th><strong>Range</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>rdf:type</td>
<td>zero-or-many</td>
<td>Resource</td>
<td>Reference</td>
<td>n/a</td>
<td>The resource type URIs. One reference should have the value of <a href="http://open-services.net/ns/asset#Artifact">http://open-services.net/ns/asset#Artifact</a>.</td>
<td></td>
</tr>
<tr>
<td>dcterms:title</td>
<td>exactly-one</td>
<td>unspecified XMLLiteral</td>
<td>n/a</td>
<td>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.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>oslc:label</td>
<td>zero-or-one</td>
<td>unspecified XMLLiteral</td>
<td>n/a</td>
<td>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.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>oslc_asset:content</td>
<td>exactly-one</td>
<td>Resource</td>
<td>Reference</td>
<td>Artifact Media</td>
<td>The media resource reference URI (the artifact bytes).</td>
<td></td>
</tr>
<tr>
<td>dcterms:format</td>
<td>zero-or-one</td>
<td>unspecified String</td>
<td>n/a</td>
<td>The mime type of the artifact media resource.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>oslc_asset:size</td>
<td>zero-or-one</td>
<td>True</td>
<td>Integer</td>
<td>n/a</td>
<td>The size of the artifact media resource in bytes.</td>
<td></td>
</tr>
<tr>
<td>dcterms:created</td>
<td>zero-or-one</td>
<td>True</td>
<td>DateTime</td>
<td>n/a</td>
<td>Timestamp of resource creation (reference: Dublin Core).</td>
<td></td>
</tr>
<tr>
<td>dcterms:modified</td>
<td>zero-or-one</td>
<td>True</td>
<td>DateTime</td>
<td>n/a</td>
<td>Timestamp last resource modification (reference: Dublin Core).</td>
<td></td>
</tr>
</tbody>
</table>
OSLC Asset Management service providers MUST support 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] as defined in OSLC Core Specifications [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] 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] as defined in OSLC Core Specification [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 `rdf:member` as defined in OSLC Core RDF/XML Examples [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].

### Delegated User Interface Dialogs

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

The service providers support the delegated Uls 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].

### Appendix B: Resource Shapes

This section is informative


### Appendix C: Notices and References

#### Contributors

- Brad Sandler (IBM)
- Dave Johnson (IBM)
- Edwin Freenenhorst (Shell)
- Eric Bordeaux (IBM)
- Gilli 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-assetmanagement-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].

#### 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). Details of the Covenant may be found here [http://open-services.net/bin/view/Main/AssetMgSpecificationV2Covenant].

#### References

- OSLC-Asset 1.0 - OSLC Asset Management Specifications 1.0 [http://open-services.net/bin/view/Main/AssetMgSpecificationV1]
- OSLC Core - OSLC Core Specification 2.0 [http://open-services.net/bin/view/Main/OslcCoreSpecification]
- FOAF - Friend of a Friend (FOAF) 1.0 98 [http://xmlns.com/foaf/spec/20100809.html]
- HTTP 1.1 - Hypertext Transfer Protocol (HTTP/1.1) [http://tools.ietf.org/html/rfc2616]
- JSON - JavaScript Object Notation [http://json.org/]