



CTS2 Value Set Services

Version 1.1

OMG Document Number: formal/2013-05-13
Standard document URL: <http://www.omg.org/spec/cts2/1.1/PDF>

Copyright © 2013, Mayo Clinic
Copyright © 2013, Object Management Group

USE OF SPECIFICATION - TERMS, CONDITIONS & NOTICES

The material in this document details an Object Management Group specification in accordance with the terms, conditions and notices set forth below. This document does not represent a commitment to implement any portion of this specification in any company's products. The information contained in this document is subject to change without notice.

LICENSES

The companies listed above have granted to the Object Management Group, Inc. (OMG) a nonexclusive, royalty-free, paid up, worldwide license to copy and distribute this document and to modify this document and distribute copies of the modified version. Each of the copyright holders listed above has agreed that no person shall be deemed to have infringed the copyright in the included material of any such copyright holder by reason of having used the specification set forth herein or having conformed any computer software to the specification.

Subject to all of the terms and conditions below, the owners of the copyright in this specification hereby grant you a fully-paid up, non-exclusive, nontransferable, perpetual, worldwide license (without the right to sublicense), to use this specification to create and distribute software and special purpose specifications that are based upon this specification, and to use, copy, and distribute this specification as provided under the Copyright Act; provided that: (1) both the copyright notice identified above and this permission notice appear on any copies of this specification; (2) the use of the specifications is for informational purposes and will not be copied or posted on any network computer or broadcast in any media and will not be otherwise resold or transferred for commercial purposes; and (3) no modifications are made to this specification. This limited permission automatically terminates without notice if you breach any of these terms or conditions. Upon termination, you will destroy immediately any copies of the specifications in your possession or control.

PATENTS

The attention of adopters is directed to the possibility that compliance with or adoption of OMG specifications may require use of an invention covered by patent rights. OMG shall not be responsible for identifying patents for which a license may be required by any OMG specification, or for conducting legal inquiries into the legal validity or scope of those patents that are brought to its attention. OMG specifications are prospective and advisory only. Prospective users are responsible for protecting themselves against liability for infringement of patents.

GENERAL USE RESTRICTIONS

Any unauthorized use of this specification may violate copyright laws, trademark laws, and communications regulations and statutes. This document contains information which is protected by copyright. All Rights Reserved. No part of this work covered by copyright herein may be reproduced or used in any form or by any means--graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems--without permission of the copyright owner.

DISCLAIMER OF WARRANTY

WHILE THIS PUBLICATION IS BELIEVED TO BE ACCURATE, IT IS PROVIDED "AS IS" AND MAY CONTAIN ERRORS OR MISPRINTS. THE OBJECT MANAGEMENT GROUP AND THE COMPANIES LISTED ABOVE MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS PUBLICATION, INCLUDING BUT NOT

LIMITED TO ANY WARRANTY OF TITLE OR OWNERSHIP, IMPLIED WARRANTY OF MERCHANTABILITY OR WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE OR USE.

IN NO EVENT SHALL THE OBJECT MANAGEMENT GROUP OR ANY OF THE COMPANIES LISTED ABOVE BE LIABLE FOR ERRORS CONTAINED HEREIN OR FOR DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL, RELIANCE OR COVER DAMAGES, INCLUDING LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY ANY USER OR ANY THIRD PARTY IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The entire risk as to the quality and performance of software developed using this specification is borne by you. This disclaimer of warranty constitutes an essential part of the license granted to you to use this specification.

RESTRICTED RIGHTS LEGEND

Use, duplication or disclosure by the U.S. Government is subject to the restrictions set forth in subparagraph (c) (1) (ii) of The Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013 or in subparagraph (c)(1) and (2) of the Commercial Computer Software - Restricted Rights clauses at 48 C.F.R. 52.227-19 or as specified in 48 C.F.R. 227-7202-2 of the DoD F.A.R. Supplement and its successors, or as specified in 48 C.F.R. 12.212 of the Federal Acquisition Regulations and its successors, as applicable. The specification copyright owners are as indicated above and may be contacted through the Object Management Group, 109 Highland Avenue, Needham, MA 02494, U.S.A.

TRADEMARKS

MDA®, Model Driven Architecture®, UML®, UML Cube logo®, OMG Logo®, CORBA® and XMI® are registered trademarks of the Object Management Group, Inc., and Object Management Group™, OMG™, Unified Modeling Language™, Model Driven Architecture Logo™, Model Driven Architecture Diagram™, CORBA logos™, XMI Logo™, CWM™, CWM Logo™, IIOP™, IMM™, MOF™, OMG Interface Definition Language (OMG IDL)™, and OMG Systems Modeling Language (OMG SysML)™ are trademarks of the Object Management Group. All other products or company names mentioned are used for identification purposes only, and may be trademarks of their respective owners.

COMPLIANCE

The copyright holders listed above acknowledge that the Object Management Group (acting itself or through its designees) is and shall at all times be the sole entity that may authorize developers, suppliers and sellers of computer software to use certification marks, trademarks or other special designations to indicate compliance with these materials.

Software developed under the terms of this license may claim compliance or conformance with this specification if and only if the software compliance is of a nature fully matching the applicable compliance points as stated in the specification. Software developed only partially matching the applicable compliance points may claim only that the software was based on this specification, but may not claim compliance or conformance with this specification. In the event that testing suites are implemented or approved by Object Management Group, Inc., software developed using this specification may claim compliance or conformance with the specification only if the software satisfactorily completes the testing suites.

Table of Contents

1. Introduction	1
1.1 Notation	1
1.2 Identifiers	1
1.3 Stereotypes	1
1.4 Read Only Attributes	2
2. Value Set Information Model	3
2.1 Value Set Catalog Entry	3
2.1.1 Class ValueSetCatalogEntry	3
2.2 Value Set Catalog and List	4
2.2.1 Class ValueSetCatalogEntryDirectory	4
2.2.2 Class ValueSetCatalogEntryList	5
2.2.3 Class ValueSetCatalogEntryListEntry	5
2.2.4 Class ValueSetCatalogEntrySummary	5
3. Value Set Services	7
3.1 Value Set Catalog Read Service	7
3.1.1 Interface ValueSetCatalogReadService	7
3.2 Value Set Catalog Query Service	9
3.2.1 Interface ValueSetCatalogQueryService	9
3.3 Value Set Catalog History Service	14
3.3.1 Interface ValueSetCatalogHistoryService	14
3.4 Value Set Catalog Maintenance Service	16
3.4.1 Interface UpdateValueSetRequest	17
3.4.2 Class ValueSetCatalogMaintenanceService	17
4. Value Set Definition Information Model	19
4.1 Value Set Definition	19
4.1.1 Class ValueSetDefinition	20
4.1.2 Class ValueSetDefinitionEntry	20
4.2 Value Set Definition List and Directory	21
4.2.1 Class ValueSetDefinitionDirectory	21
4.2.2 Class ValueSetDefinitionDirectoryEntry	22
4.2.3 Class ValueSetDefinitionList	22
4.2.4 Class ValueSetDefinitionListEntry	22
4.3 Value Set Definition Entry	23

4.3.1 Class AssociatedEntitiesReference	23
4.3.2 Class CompleteCodeSystemReference	24
4.3.3 Class CompleteValueSetReference	24
4.3.4 Class ExternalValueSetDefinition	25
4.3.5 Class PropertyQueryReference	25
4.3.6 Class SpecificEntityList	26
4.4 Value Set Resolution	27
4.4.1 Class IterableResolvedValueSet	28
4.4.2 Class Resolved ValueSet	28
5. Value Set Definition Services	31
5.1 Value Set Definition Read Services	31
5.1.1 Interface ValueSetDefinitionReadService	31
5.2 Value Set Definition Query Services	34
5.2.1 Interface ValueSetDefinitionQueryService	34
5.3 Value Set History Service	37
5.3.1 Interface ValueSetDefinitionHistoryService	37
5.4 Value Set Definition Maintenance Services	39
5.4.1 Interface ValueSetDefinitionMaintenanceService	39
6. Value Set Definition Resolution Services	49
6.1 Value Set Resolution Services	49
6.1.1 Interface ValueSetDefinitionResolution	49
7. Resolved Value Set Services	55
7.1 Resolved Value Set Resolution Service	55
7.1.1 Interface ResolvedValueSetResolution	55
7.2 ResolvedValueSetLoader	59
7.2.1 Interface ResolvedValueSetLoader	59
7.3 Resolved Value Set Query Service	60
7.3.1 Interface ResolvedValueSetQueryService	61

1 Introduction

The CTS2 specification is divided into a number of separably implementable profiles. This section on Value Set Services describes how collections of `EntityReferences` are described, defined, resolved, and disseminated. A value set represents a collection of entities (classes, properties and/or individuals) drawn from one or more code systems. While value sets are often used to provide “pick lists” and validation rules for fields in data bases, messages and forms, they can be used in any circumstance where a collection of entities is required.

1.1 Notation

Model elements are referenced using `Typewriter` font.

1.2 Identifiers

- Class names are capitalized camel case (`CodeSystemCatalog`, `String`).
- Attribute and role names are lower camel case (`version`, `sourceAndRole`).
- Enumeration values are all capital letters with underscores ('_') introduced as needed for clarity. (`EMPTY`, `STOP_ON_ERROR`).
- Method names are `lowerCamelCase`.

1.3 Stereotypes

The following stereotypes are used throughout the model:

- **dataType** - UML `dataType` semantics.
- **enumeration** - UML enumeration semantics.
- **interface** - UML interface semantics.
- **mixin** - a class that provides a certain functionality to be inherited by a subclass, while not meant for instantiation.
- **opt** - an optional parameter in a method call. This stereotype should always be accompanied by a 0..1 cardinality and is provided to make optionality visible in the modeling tool.
- **optparam** - an optional attribute in a class instance. This indicates that an implementation must be able to distinguish three alternatives: (1) add or change the attribute to whatever is in the class, (2) remove the attribute if its minimum cardinality is zero, and (3) leave the attribute unchanged.
- **delta** - a method call that changes the state of the service. Note that the semantics of this stereotype is the inverse of the `is query model attribute`.
- **exception** - an exception.
- **exceptionSet** - a set of exceptions, the members of which are represented by aggregation.

1.4 Read Only Attributes

The CTS2 PIM is based on a RESTful architectural style. Resources can be created, updated, read, and/or deleted, and every resource has one or more immutable characteristics - characteristics that, if changed, would change the identity of the resource itself. In the information model we distinguish these characteristics from those that can be modified by declaring the identifying characteristics as *ReadOnly*. The computational model utilizes a pattern based on this information where the *create* operation(s) supply the identifying characteristics plus any of the non-optional mutable characteristics. The *update* operations then supply a unique identifier and a set of one or more mutable characteristics to be changed.

2 Value Set Information Model

2.1 Value Set Catalog Entry

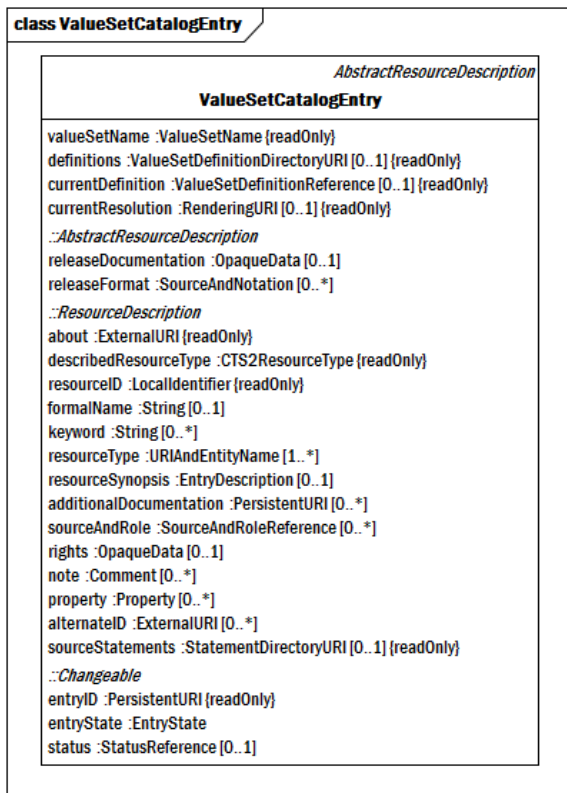


Figure 2.1 - Value Set Catalog Entry

A `ValueSetCatalogEntry` carries information about the creators, distributors, purpose, use, etc. about a value set. The catalog does not carry the actual definition of a value set as (a) this may vary over time and (b) it is possible for more than one definition to be in use at any given point in time.

2.1.1 Class ValueSetCatalogEntry

An entry in a value set catalog that describes the purpose, use, etc. of a value set.

Superclasses

- Every instance of `ValueSetCatalogEntry` is also an instance of `AbstractResourceDescription`.

Attributes

- **valueSetName** - the unique name of this value set within the context of the CTS2 service instance.
- **definitions** - a `DirectoryURI` that references a set of definition (versions) for this value set. If the service profile does not support value set definitions, this attribute should be omitted. If the service supports definitions but none are

known for this value set, the parameter should be supplied, but will resolve to an empty directory.

- **currentDefinition** - a reference to the definition associated with the CURRENT tag, if any.
- **currentResolution** - a URI that resolves to the resolution of this value set using the CURRENT tag for both the definition and the code system versions.

2.2 Value Set Catalog and List

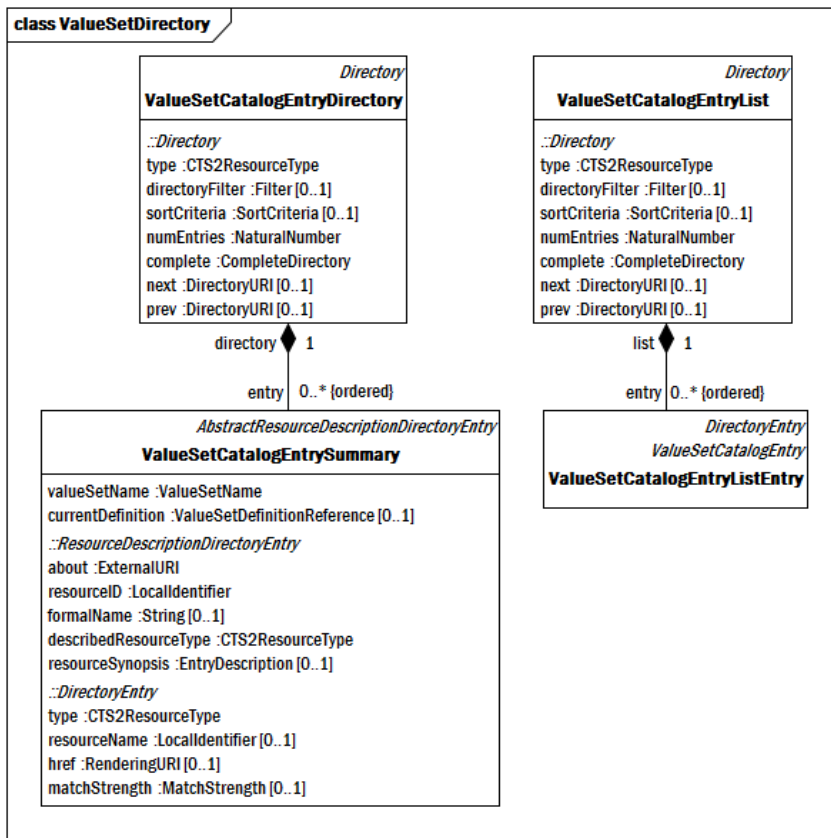


Figure 2.2 - Value Set Directory

A ValueSetCatalog contains a summary of value sets that meet a specified criteria.

2.2.1 Class ValueSetCatalogEntryDirectory

A catalog that summarizes a set of value sets known to the service.

Superclasses

- Every instance of ValueSetCatalogEntryDirectory is also an instance of Directory.

Attributes

- **entry** - an entry in the directory.

Invariants

1. type must be `VALUE_SET`.
2. All entries in the directory must be value sets.

2.2.2 Class `ValueSetCatalogEntryList`

An iterable collection of `ValueSetCatalogEntries`.

Superclasses

- Every instance of `ValueSetCatalogEntryList` is also an instance of `Directory`.

Attributes

- **entry** - an entry in the list

2.2.3 Class `ValueSetCatalogEntryListEntry`

A `ValueSetCatalogEntry` as it appears in a directory listing.

Superclasses

- Every instance of `ValueSetCatalogEntryListEntry` is also an instance of `ValueSetCatalogEntry`.
- Every instance of `ValueSetCatalogEntryListEntry` is also an instance of `DirectoryEntry`.

Invariants

1. The type must be `VALUE_SET`.

2.2.4 Class `ValueSetCatalogEntrySummary`

A synopsis of a `ValueSet` along with information about how to access the complete resource.

Superclasses

- Every instance of `ValueSetCatalogEntrySummary` is also an instance of `AbstractResourceDescriptionDirectoryEntry`.

Attributes

- **valueSetName** - the local identifier assigned to the value set within the context of the service.
- **currentDefinition** - the name and URI of the value set definition that is associated with the `CURRENT` tag for this value set.

Invariants

1. The type must be `VALUE_SET`.

3 Value Set Services

3.1 Value Set Catalog Read Service

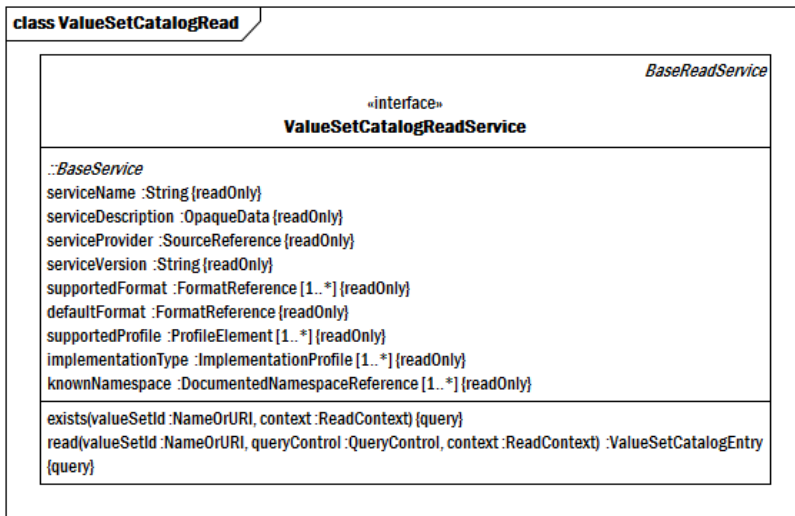


Figure 3.1 - Value Set Catalog Read

The `ValueSetCatalogReadService` provides direct access to the service value set catalog. Entries in the catalog can be retrieved either by an identifying URI or by a locally unique name.

3.1.1 Interface `ValueSetCatalogReadService`

Read a `ValueSetCatalogEntry` by name or URI.

Superclasses

- Every instance of `ValueSetCatalogReadService` is also an instance of `BaseReadService`.

3.1.1.1 Operation: `exists`

Determine if a catalog entry for the named value is known to the service.

Input Parameters

- **valueSetId** - the local name, about URI, OID or alternateID of the resource to check for existence (Type: `NameOrURI`).
- **context_{OPT}** - parameters that control the language, date, and other contextual variables (Type: `ReadContext`).

Return Type: `Boolean`

Exceptions

- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.

- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.

3.1.1.2 Operation: read

Retrieve the specified value set.

Input Parameters

- **valueSetId** - the local name, about URI, OID or alternateID of the resource to read (Type: `NameOrURI`)
- **queryControl_{OPT}** - parameters that control the ordering, timeout, and query behavior. (Type: `QueryControl`)
- **context_{OPT}** - parameters that control the language, date, and other contextual variables. (Type: `ReadContext`)

Return Type: ValueSetCatalogEntry

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The `format` is not supported by the service implementation.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.

3.2 Value Set Catalog Query Service

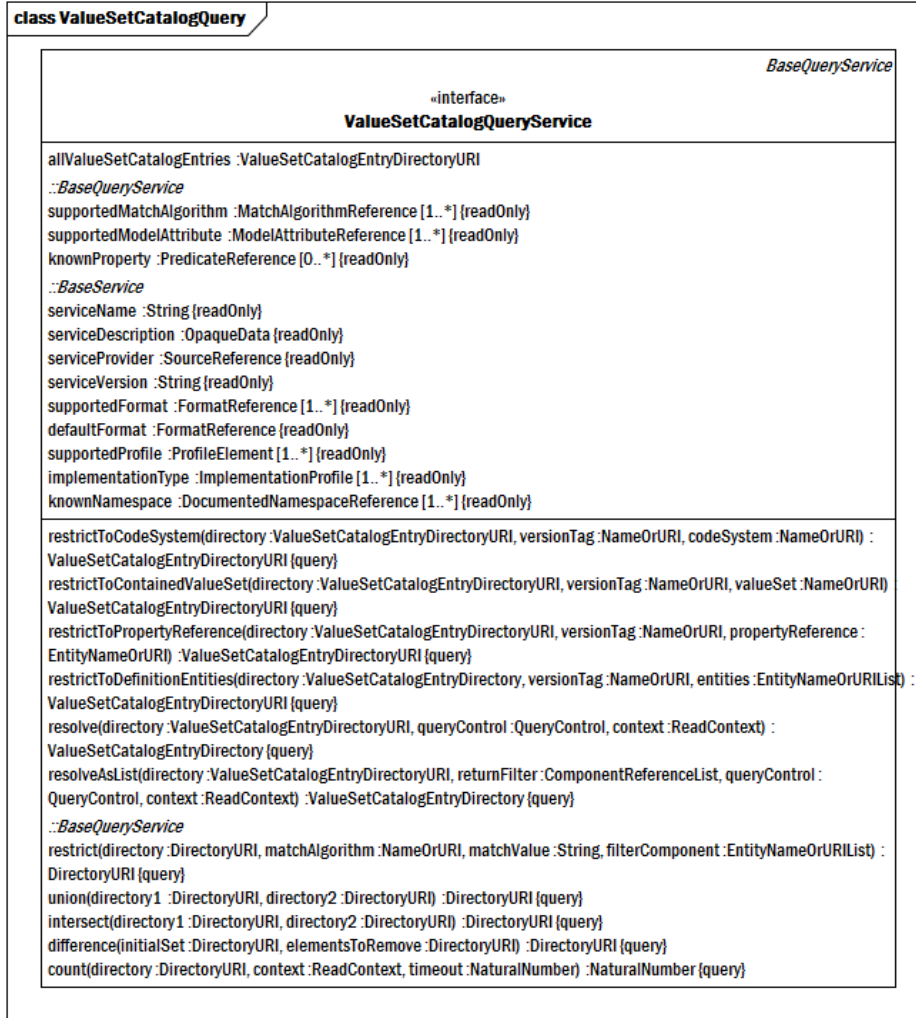


Figure 3.2 - Value Set Catalog Query

3.2.1 Interface ValueSetCatalogQueryService

A service that allows the selection and enumeration of ValueSetCatalogEntries known to CTS2 service implementation.

Superclasses

- Every instance of ValueSetCatalogQueryService is also an instance of BaseQueryService.

Attributes

- **allValueSetCatalogEntries** - a `DirectoryURI` that resolves to the set of `ValueSetCatalogEntries` known to the service instance.

3.2.1.1 Operation: restrictToCodeSystem

Return a reference to the subset of the supplied value sets whose definitions reference the named code system.

Input Parameters

- **directory** - a `DirectoryURI` that resolves to a set of value set catalog entries (Type: `ValueSetCatalogEntryDirectoryURI`).
- **versionTag**_{OPT} - the name or URI of a version tag that will be used to select the correct value set definition. If not supplied, the value set definition with the `CURRENT` version tag will be used. (Type: `NameOrURI`)
- **codeSystem** - the name or URI of a code system to use for restriction (Type: `NameOrURI`).

Return Type: ValueSetCatalogEntryDirectoryURI

Exceptions

- **UnknownCodeSystem** - The referenced `CodeSystem` is not recognized by the service.
- **UnsupportedVersionTag** - The `versionTag` is not recognized by the service.
- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.

3.2.1.2 Operation: restrictToContainedValueSet

Return a reference to the subset of the supplied value sets whose definitions include the supplied value set.

Input Parameters

- **directory** - a `DirectoryURI` that resolves to a set of `ValueSetCatalogEntries` (Type: `ValueSetCatalogEntryDirectoryURI`).
- **versionTag**_{OPT} - the name or URI of a version tag that will be used to select the correct value set definition. If not supplied, the value set definition with the `CURRENT` version tag will be used. (Type: `NameOrURI`)
- **valueSet** - the name or URI of the filter value set (Type: `NameOrURI`).

Return Type: ValueSetCatalogEntryDirectoryURI

Exceptions

- **UnsupportedVersionTag** - The `versionTag` is not recognized by the service.
- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **InvalidDirectoryURI** - The supplied directory URI is not valid.

- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.

3.2.1.3 Operation: restrictToPropertyReference

Return a reference to the subset of the supplied value sets whose definitions include a reference to the named property predicate.

Input Parameters

- **directory** - A reference to the set of catalog entries to restrict. (Type: ValueSetCatalogEntryDirectoryURI)
- **versionTag_{OPT}** - The name or URI of a version tag that will be used to select the correct value set definition. If not supplied, the value set definition with the CURRENT version tag will be used. (Type: NameOrURI)
- **propertyReference** - The name or URI of the property reference to test for. (Type: EntityNameOrURI)

Return Type: ValueSetCatalogURI

Exceptions

- **UnsupportedVersionTag** - The versionTag is not recognized by the service.
- **UnknownPredicateNamespaceName** - The namespace in the predicate identifier is not known to or supported by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.

3.2.1.4 Operation: restrictToDefinitionEntities

Return a reference to the subset of the supplied value sets whose definitions directly reference one or more of the referenced entities. Note that this query only returns definitions that include the entities in either the SpecificEntityList or the AssociatedEntitiesReference variations. It does not include entities that would be include in the resolution of the definition nor entities that are referenced indirectly via the CompleteValueSetReference variant.

Input Parameters

- **directory** - A reference to the set of catalog entries to restrict. (Type: ValueSetCatalogEntryDirectoryURI)
- **versionTag_{OPT}** - The name or URI of a version tag that will be used to select the correct value set definition. If not supplied, the value set definition with the CURRENT version tag will be used. (Type: NameOrURI)
- **entities** - The set of entities to use as a filter. (Type: EntityNameOrURIList)

Return Type: ValueSetCatalogEntryDirectoryURI

Exceptions

- **UnsupportedVersionTag** - The versionTag is not recognized by the service.

- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnknownEntity** - The `EntityNameOrURI` is not known to the service.

3.2.1.5 Operation: resolve

Resolve a `ValueSetCatalogDirectoryURI` to a set of `ValueSetCatalogSummaryEntries`.

Input Parameters

- **directory** - a `DirectoryURI` that resolves to a set of `ValueSetCatalogEntries` (Type: `ValueSetCatalogEntryDirectoryURI`).
- **queryControl**_{OPT} - parameters that control the ordering, timeout, and query behavior (Type: `QueryControl`).
- **context**_{OPT} - parameters that control the language, date, and other contextual variables (Type: `ReadContext`).

Return Type: ValueSetCatalogEntryDirectory

Exceptions

- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The `format` is not supported by the service implementation.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.

3.2.1.6 Operation: resolveAsList

Resolve a `ValueSetCatalogDirectoryURI` to a set of `ValueSetCatalogEntries`.

Input Parameters

- **directory** - a `DirectoryURI` that references a set of `ValueSetCatalogEntries` (Type: `ValueSetCatalogEntryDirectoryURI`).
- **returnFilter**_{OPT} - a list of zero or more component references. If present, the returned list entries will contain only the required elements and attributes plus any elements or attributes named in the filter. If absent all elements and attributes will be returned. (Type: `ComponentReferenceList`).
- **queryControl**_{OPT} - parameters that control the ordering, timeout, and query behavior (Type: `QueryControl`).
- **context**_{OPT} - parameters that control the language, date, and other contextual variables (Type: `ReadContext`).

Return Type: `ValueSetCatalogEntryDirectory`

Exceptions

- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The `format` is not supported by the service implementation.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.

3.3 Value Set Catalog History Service

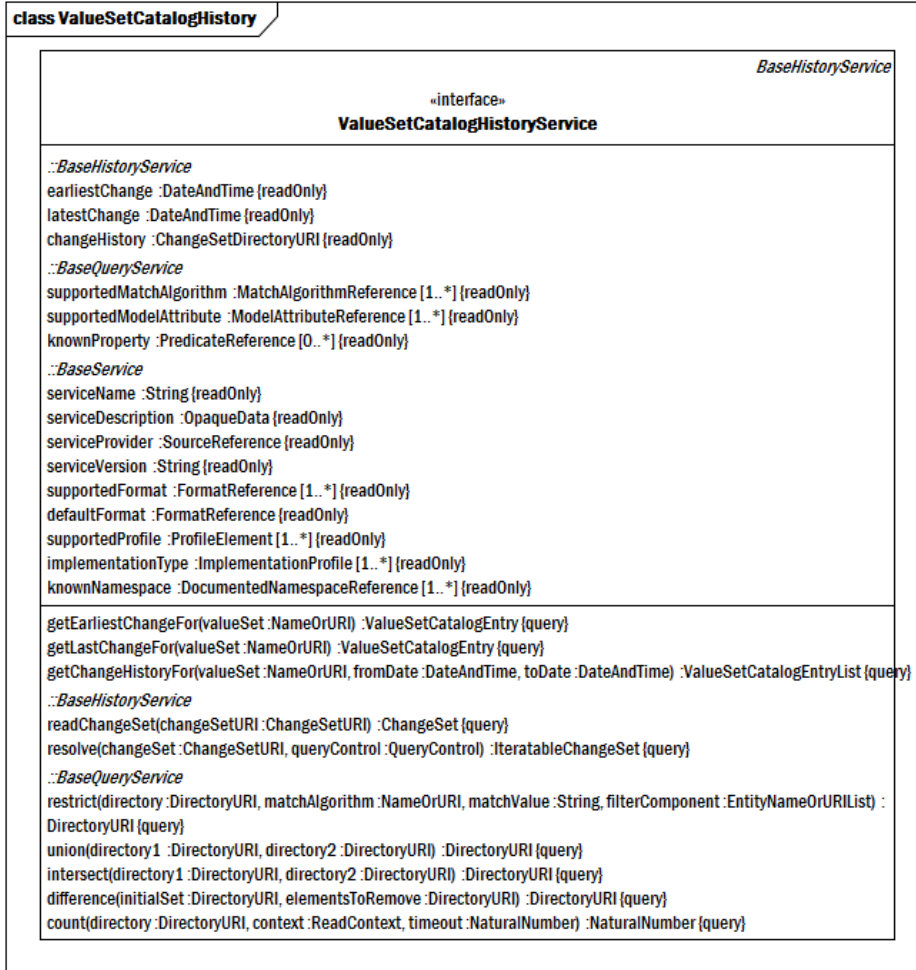


Figure 3.3 - Value Set Catalog History

The `ValueSetCatalogHistoryService` allows the client to find the first change, last change, and the set of changes that occurred to a `ValueSetCatalogEntry` in a specified time interval.

3.3.1 Interface ValueSetCatalogHistoryService

Superclasses

- Every instance of `ValueSetCatalogHistoryService` is also an instance of `BaseHistoryService`.

3.3.1.1 Operation: getEarliestChangeFor

Return the first known state of the specified `ValueSetCatalogEntry`.

Input Parameters

- **valueSet** - the name or URI of a value set (Type: NameOrURI).

Return Type: ValueSetCatalogEntry

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.

3.3.1.2 Operation: getLastChangeFor

Return the last known change to the specified ValueSetCatalogEntry.

Input Parameters

- **valueSet** - Return the earliest known change for the named value set catalog entry. (Type: NameOrURI)

Return Type: ValueSetCatalogEntry

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.

3.3.1.3 Operation: getChangeHistoryFor

Return a list of ValueSetCatalogEntries that reflect the various states that the ValueSetCatalogEntry went through between fromDate and toDate.

Input Parameters

- **valueSet** - the name or URI of a value set (Type: NameOrURI).
- **fromDate_{OPT}** - the starting date. If not supplied, states beginning with the first known state will be returned. If later than toDate, the return order is earliest to latest, otherwise it is latest to earliest. (Type: DateAndTime)
- **toDate_{OPT}** - the ending date. If not supplied, states up to the last known state will be returned. If later than fromDate, the return order is latest to earliest, otherwise it is earliest to latest. (Type: DateAndTime)

Return Type: ValueSetCatalogEntryList

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.

3.4 Value Set Catalog Maintenance Service

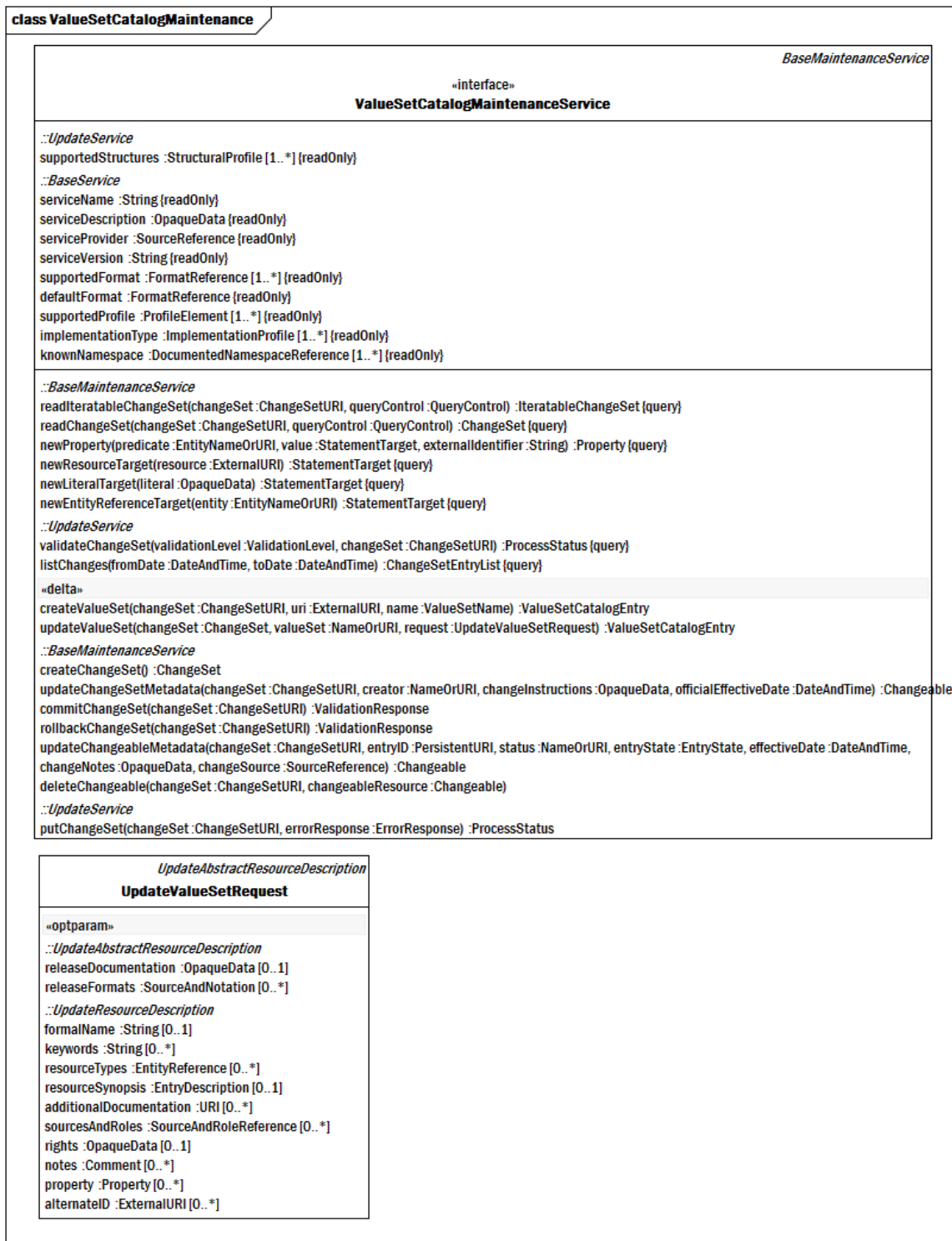


Figure 3.4 - Value Set Catalog Maintenance Service

The `ValueSetCatalogMaintenanceService` provides services to create, update, or remove `ValueSetCatalogEntries` in the service catalog.

3.4.1 Interface `UpdateValueSetRequest`

Superclasses

- Every instance of `UpdateValueSetRequest` is also an instance of `UpdateAbstractResourceDescription`.

3.4.2 Class `ValueSetCatalogMaintenanceService`

Superclasses

- Every instance of `ValueSetCatalogMaintenanceService` is also an instance of `BaseMaintenanceService`.

3.4.2.1 Operation: `createValueSet`

Create a new `ValueSetCatalogEntry`.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an open change set to record the change. (Type: `ChangeSetURI`)
- **uri** - an external URI that uniquely identifies the new value set. (Type: `ExternalURI`)
- **name** - a local identifier that uniquely identifies the value set within the service context. (Type: `ValueSetName`)

Return Type: `ValueSetCatalogEntry`

Exceptions

- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **DuplicateValueSetURI** - The supplied value set URI already exists as an about or alternate ID for a value set in the service.
- **DuplicateValueSetName** - The supplied `valueSetName` already exists in the service.

3.4.2.2 Operation: `updateValueSet`

Update an existing `ValueSetCatalogEntry`.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the name or URI of an `OPEN` change set to record the change. (Type: `ChangeSet`)
- **valueSet** - the name or URI that references an existing `ValueSetCatalogEntry`. (Type: `NameOrURI`)

- **request** - the parameters to update. (Type: UpdateValueSetRequest)

Return Type: ValueSetCatalogEntry

Exceptions

- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **UnsupportedOntologyLanguage** - The supplied ontology language is not supported by the service.
- **UnsupportedStatus** - The name or URI of the `Changeable` `status` property is not recognized by the service.
- **UnsupportedSource** - The supplied source is not recognized by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedRole** - The role name or URI is not recognized by the service.
- **UnsupportedOntologySyntax** - The supplied ontology syntax is not supported by the service.

4 Value Set Definition Information Model

4.1 Value Set Definition



Figure 4.1 - Value Set Definition

A `ValueSetDefinition` describes the rules that determine which entity references (value meanings) belong to a value set at a given point in time. The definition of what belongs in a value set can evolve over time, and it is possible for there to be multiple definitions active at any given point in time - perhaps one for a system in general use, a second for a newer system, and a third for testing. A `ValueSetDefinition` may or may not identify a specific version of a code system. The decision of which version is to be used depends on the context and needs of the community.

`ValueSetDefinition` and the supporting model has been designed to allow multiple variations on the “binding” of definitions to value sets, code system versions to definitions, and the combination to concept domains.

4.1.1 Class ValueSetDefinition

Superclasses

- Every instance of `ValueSetDefinition` is also an instance of `ResourceVersionDescription`.

Attributes

- **definedValueSet** - a reference to the value set being defined.
- **versionTag** - a version tag assigned to this definition by the implementing service.
- **entry** - A component in a value set definition.

4.1.2 Class ValueSetDefinitionEntry

An element of a value set definition that, when resolved yields a set of entity references that are to be included in, excluded from or intersected with the set of elements that represent the full resolution of the definition.

Note that only `ACTIVE` entity references are included. `INACTIVE` entity references may never be considered for inclusion or exclusion in the resolution of a value set definition.

Attributes

- **operator** - instructions for how the results of the entry evaluation will be applied to the set of entities in the base definition. Results can be added to the set (`UNION`), intersected with members already in the set (`INTERSECT`), or be removed from the set (`DIFFERENCE`).
- **entryType** - the discriminant that determines the definition type.

4.1.2.1 Enum SetOperator

General set operators that specify set union, difference, or intersection in various contexts.

Enumeration Types:

- **UNION** - Include all members of both sets in the result.
- **INTERSECT** - include only members common to both sets in the result.
- **SUBTRACT** - remove all members in the second set from the first set.

4.2 Value Set Definition List and Directory

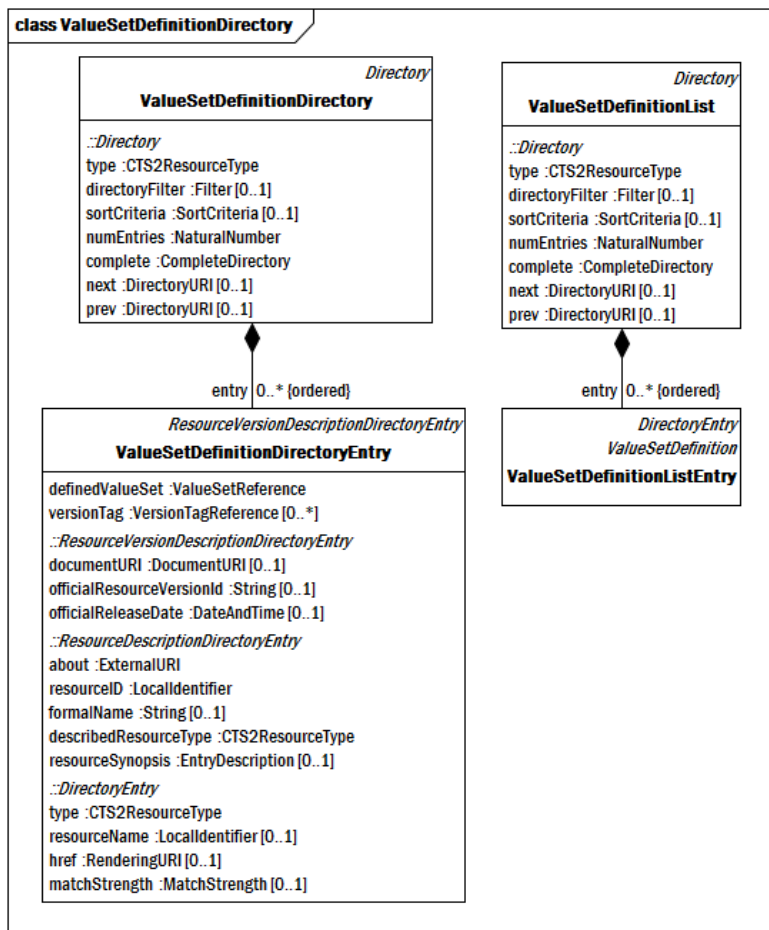


Figure 4.2 - ValueSet Definition Directory

4.2.1 Class ValueSetDefinitionDirectory

A directory of ValueSetDefinition resources that meet a specified criteria.

Superclasses

- Every instance of ValueSetDefinitionDirectory is also an instance of Directory.

Attributes

- entry - an entry in a ValueSetDefinitionDirectory

4.2.2 Class ValueSetDefinitionDirectoryEntry

A synopsis of a ValueSetDefinition along with information about how to access the complete resource.

Superclasses

- Every instance of ValueSetDefinitionDirectoryEntry is also an instance of ResourceVersionDescriptionDirectoryEntry.

Attributes

- **definedValueSet** - the value set that this definition applies to.
- **versionTag** - a version tag assigned to this entry by the service instance.

4.2.3 Class ValueSetDefinitionList

A collection of complete ValueSetDefinition resources that meet a specified criteria.

Superclasses

- Every instance of ValueSetDefinitionList is also an instance of Directory.

Attributes

- **entry** - an entry in a ValueSetDefinitionList.

4.2.4 Class ValueSetDefinitionListEntry

An instance of ValueSetDefinition that meets a specified filter criteria.

Superclasses

- Every instance of ValueSetDefinitionListEntry is also an instance of ValueSetDefinition.
- Every instance of ValueSetDefinitionListEntry is also an instance of DirectoryEntry.

4.3 Value Set Definition Entry

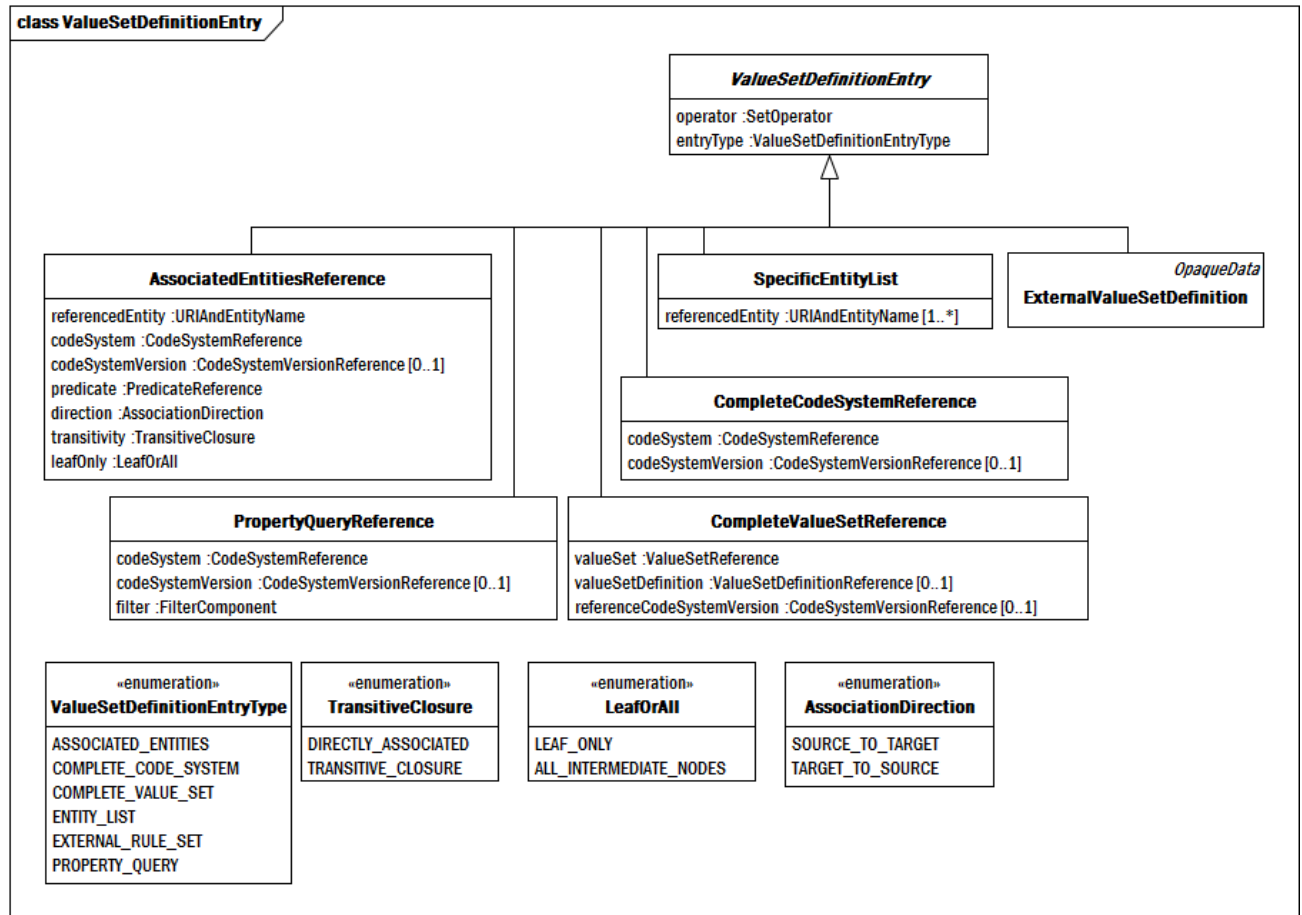


Figure 4.3 - ValueSet Definition Entry Model

4.3.1 Class AssociatedEntitiesReference

The description of a set of entities that are associated with a referenced entity. This description names a reference entity and an association predicate, which identifies a set of entities that are related to the reference entity according to a given code system. The description can reference the direct targets of the association (children), the direct sources of the association (parents), the transitive closure of the association targets (descendants), the transitive closure of the association sources (ancestors) and can state whether all intermediate nodes are included in the closure or just the leaf nodes.

Note that the terms “parent” and “children” are asserted in reference to the predicate itself. As an example, in the association “arm subclassOf bodyPart,” the “parent” is arm and the “child” is bodyPart.

Superclasses

- Every instance of `AssociatedEntitiesReference` is also an instance of `ValueSetDefinitionEntry`.

Attributes

- **referencedEntity** - the entity reference that is the root of the association description. `referencedEntity` is not considered to be part of the resolution of `AssociatedEntitiesReference`. If it is to be included in the final set, it must be explicitly named in a second `SpecificEntityList` construct.
- **codeSystem** - the code system that makes the association assertions.
- **codeSystemVersion** - the version of the code system that makes the association assertions. If present, `codeSystemVersion` must be a version of `codeSystem`. If this attribute is present, the referenced version of the code system will always be used to resolve the associations. If absent, the specific version of the code system to be used in resolution is determined in the resolve value set definition call itself.
- **predicate** - the association predicate to be used in resolving the definition entry.
- **direction** - an indicator that determines whether the `referencedEntity` serves in the source (`SOURCE_TO_TARGET`) or target (`TARGET_TO_SOURCE`) role.
- **transitivity** - an indicator that determines whether only entity references that participate in a direct association with the `referencedEntity` are considered to be a part of the definition or whether all entity references that appear on the transitive closure of entity reference are to be included. Note that `leafOnly` further qualifies this result.
- **leafOnly** - an indicator that determines whether all nodes in the transitive closure are to be included in the resulting set or only the nodes that terminate the closure. This indicator is ignored if transitivity is not `TRANSITIVE_CLOSURE`. Note that nodes that participate in cycles are not included if `leafOnly` is set to `LEAF_ONLY`.

4.3.2 Class `CompleteCodeSystemReference`

An entry that, when resolved, returns all of the active entity references in a given code system. This includes all entity references that appear as the source of one or more statements in the code system, whether the assertions are made directly by a version of the code system or indirectly by a version of a different code system that is imported. Note that targets are not included to prevent codes from `rdf`, `rdfs`, `owl`, etc. being included in this resolution set.

Superclasses

- Every instance of `CompleteCodeSystemReference` is also an instance of `ValueSetDefinitionEntry`.

Attributes

- **codeSystem** - A reference to the code system whose codes are to be included.
- **codeSystemVersion** - A reference to the specific version of the code system to include. If not supplied, the specific version of the code system is determined in the resolution call itself.

4.3.3 Class `CompleteValueSetReference`

A reference to a value set that, when resolved, results in a set of entity references that are included in this entry. An entry of this type can just name a value set, meaning that the specific definition is determined in the resolve value set call, can name both a value set and value set definition, meaning that the specific definition is always used in the resolution. It can also specify one or more code system versions to be used in the resolution of the named value set.

Superclasses

- Every instance of `CompleteValueSetReference` is also an instance of `ValueSetDefinitionEntry`.

Attributes

- **valueSet** - a reference to the value set whose definition supplies a set of entity references.
- **valueSetDefinition** - a reference to a particular definition of `valueSet` that is to be used in resolving this reference. If absent, the specific definition is determined by the resolve value set call. If present, `valueSetDefinition` must be a definition of `valueSet` and will always be used to resolve this entity set.
- **referenceCodeSystemVersion** - a reference to a `CodeSystemVersion` that will be used to resolve this call. `referenceCodeSy` will only be used if one or more components of the resolution of value Set identify a code system without specifying a specific version. At most, only one version of a given code system may appear in the `referenceCodeSystemVersion` list. While CTS2 service implementations must resolve resolution calls for definitions that carry unused reference Code SystemV entries, they may choose to issue a warning at the time the definition is created or loaded.

4.3.4 Class ExternalValueSetDefinition

A definition of a value set whose format and semantics is specified outside of the core CTS2 specification. If a given CTS2 service recognizes the syntax and semantics of this definition, it may call the appropriate process to resolve it. If the definition is not recognized, a CTS2 service implementation must not process the containing value set definition and, instead, return an error.

Superclasses

- Every instance of `ExternalValueSetDefinition` is also an instance of `ValueSetDefinitionEntry`.
- Every instance of `ExternalValueSetDefinition` is also an instance of `OpaqueData`.

4.3.5 Class PropertyQueryReference

A description of a set of entity references that are determined by applying a filter to the attribute(s) or property(s) that appear in an `EntityDescription` in a specified code system.

Superclasses

- Every instance of `PropertyQueryReference` is also an instance of `ValueSetDefinitionEntry`.

Attributes

- **codeSystem** - The code system that contains the assertions that form the attributes or properties to be tested.
- **codeSystemVersion** - The version of the code system that makes the assertions. If present, `codeSystemVersion` must be a version of `codeSystem`. If this attribute is present, the referenced version of the code system will always be used to resolve the attributes or properties. If absent, the specific version of the code system to be used in resolution is determined in the resolve value set definition call itself.
- **filter** - The filter to be applied to entities in the referenced code system.

4.3.6 Class `SpecificEntityList`

A list of specific entity references that are to be included in the definition. When specified in this form, the service must include all entities in this list whether they are known to the service or not, and whether they are currently `ACTIVE` or not.

Superclasses

- Every instance of `SpecificEntityList` is also an instance of `ValueSetDefinitionEntry`.

Attributes

- **referencedEntity** - the namespace / name or URI of an entity to be included.

4.3.6.1 Enum `AssociationDirection`

An indicator that determines whether an entity reference / predicate combination is to be evaluated with the entity reference in the source (left hand side) position of the association query or the target (right hand side or “object”) position.

Note: `AssociationDirection` doesn’t strictly belong in this diagram, but it is kind of an orphan class and we didn’t want to create an entire new diagram just for one element.

Enumeration Types:

- **SOURCE_TO_TARGET** - the statement is to be resolved with the entity reference in the role of source.
- **TARGET_TO_SOURCE** - the statement is to be resolved with the entity reference in the role of target.

4.3.6.2 Enum `LeafOrAll`

Indicates whether intermediate nodes in a transitive closure are to be included in the result or only the “leaf” nodes.

Enumeration Types:

- **LEAF_ONLY** - only nodes who can be inferred to participate as a source but not a target of an association if `direction` is `SOURCE_TO_TARGET`, or a target but not a source if `direction` is `TARGET_TO_SOURCE`, are included in the set of entity references.
- **ALL_INTERMEDIATE_NODES** - all (active) nodes are to be included in the transitive closure.

4.3.6.3 Enum `TransitiveClosure`

Indicates whether only directly associated nodes or the transitive closure over the supplied predicate is to be used in the resolution of an entity reference and predicate.

Enumeration Types:

- **DIRECTLY_ASSOCIATED** - only entity references that are explicitly asserted to be directly associated with the source entity are included. Associations whose `derivation` is `INFERRED` are not considered when this setting is used.

- **TRANSITIVE_CLOSURE** - both entity references that are explicitly asserted to be directly associated with the source entity are included and entities that can be inferred to participate in the association named by predicate are included as candidates for the set, although the final resolution may depend on other factors such as a `LeafOrAll` setting.

4.3.6.4 Enum ValueSetDefinitionEntryType

A discriminant that determines the definition entry type. PSMs that support strong typing systems may implement this through the built-in type system itself rather than providing a specific element.

Enumeration Types:

- **ASSOCIATED_ENTITIES** - the definition entry is of type `AssociatedEntitiesReference`.
- **COMPLETE_CODE_SYSTEM** - the definition entry is of type `CompleteCodeSystemReference`.
- **COMPLETE_VALUE_SET** - the definition entry is of type `CompleteValueSetReference`.
- **ENTITY_LIST** - the definition entry is of type `SpecificEntityList`.
- **EXTERNAL_RULE_SET** - the definition entry is of type `ExternalValueSetDefinition`.
- **PROPERTY_QUERY** - the definition entry is of type `PropertyQueryReference`.

4.4 Value Set Resolution

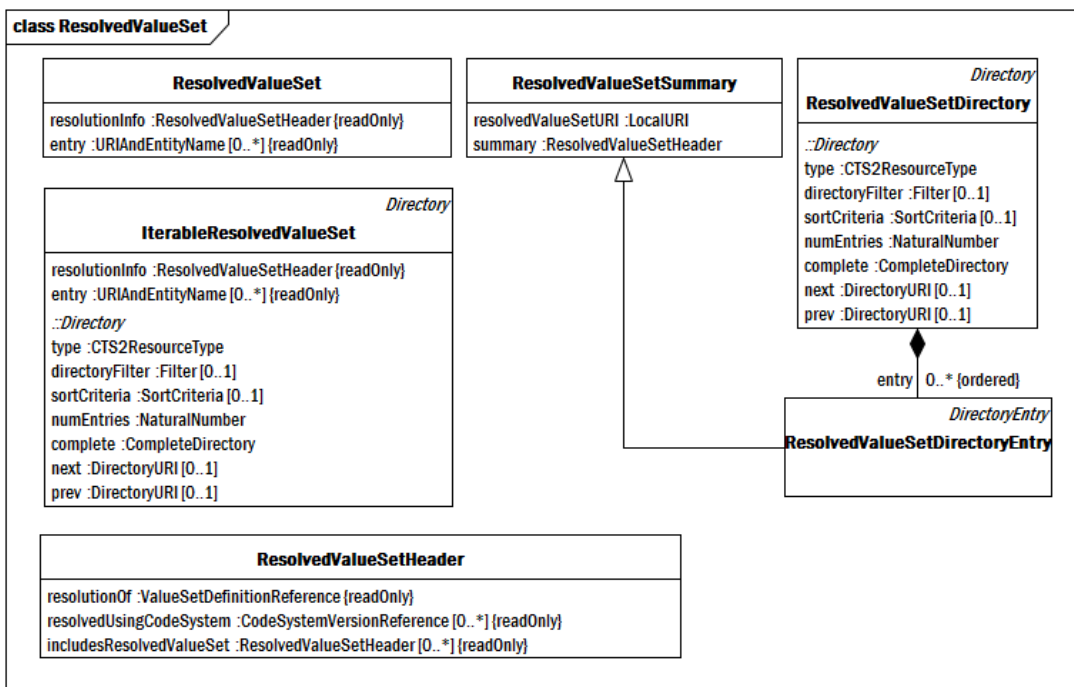


Figure 4.4 - Resolved ValueSet

ValueSet definitions can be resolved dynamically on an as needed basis, but they also can be transferred between systems without the necessity of maintaining complete copies of the code system versions that are used to resolve them. This model defines the output of a value set resolution request. It contains sufficient information that the request can be completely reproduced against a service instance that mirrors the state of the system doing the resolution. It also carries the result of the resolution that can be transferred and loaded into a system that does not mirror the source system state.

4.4.1 Class IterableResolvedValueSet

A ResolvedValueSet whose contents are available as a set of directory entries that allows iteration.

Superclasses:

- Every instance of IterableResolvedValueSet is also an instance of Directory.

Attributes:

- **resolutionInfo** - The value set definition and code system version(s) that were used to resolve the entries in this directory
- **entry** - a synopsis of the EntityDescription that has been determined to be a member of the value set named in resolutionInfo according to the stated criteria or “bindings.”

4.4.2 Class Resolved ValueSet

The result of resolving a specific value set definition against a known set of code system versions. ResolvedValueSet consists of a header that carries sufficient information that the resolution operation would be repeatable and the result of the resolution, which consists of set of EntityReferences.

Attributes

- **resolutionInfo** - the set of parameters that were used in the resolution of this set. resolution Info provides all the information necessary to recreate the actual resolution when presented to a CTS2 service instance that contains all of the necessary definitions and code system versions.
- **entry** - a synopsis of the EntityDescription that has been determined to be a member of the value set named in resolutionInfo according to the stated criteria or “bindings.”

4.4.2.1 Class Resolved ValueSetDirectory

A directory of resolved value sets.

Superclasses

- Every instance of ResolvedValueSetDirectory is also an instance of Directory.

Attributes

- **entry** - an entry in a directory of resolved value sets.

4.4.2.2 Class Resolved ValueSetDirectoryEntry

An entry in a ResolvedValueSetDirectory.

Superclasses

- Every instance of `ResolvedValueSetDirectoryEntry` is also an instance of `DirectoryEntry`.
- Every instance of `ResolvedValueSetDirectoryEntry` is also an instance of `ResolvedValueSetSummary`.

4.4.2.3 Class `ResolvedValueSetHeader`

The information required to completely resolve a value set definition.

Attributes

- **resolutionOf** - the specific value set definition that was resolved.
- **resolvedUsingCodeSystem** - a reference to a code system version that was used in the resolution of this value set definition.
- **includesResolvedValueSet** - If the value set includes other value sets, this contains the resolution information used to resolve the inner value sets.

4.4.2.4 Class `ResolvedValueSetSummary`

The local URI and a summary of a resolved value set that has been loaded into a CTS2 service instance.

Attributes

- **resolvedValueSetURI** - the URI that the value set is known by locally.
- **summary** - the information about the resolved set.

5 Value Set Definition Services

5.1 Value Set Definition Read Services

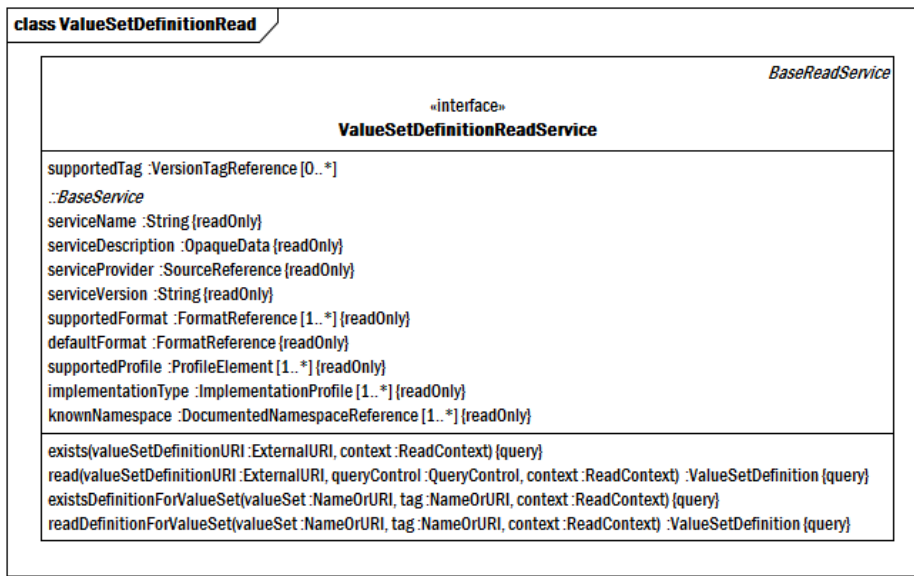


Figure 5.1 - Value Set Definition Read Service

5.1.1 Interface ValueSetDefinitionReadService

Superclasses

- Every instance of **ValueSetDefinitionReadService** is also an instance of **BaseReadService**.

Attributes

- **supportedTag** - a version tag that is known to the service. All services must support the “CURRENT” tag.

5.1.1.1 Operation: exists

Determine if the specified value set definition exists on the service.

Input Parameters

- **valueSetDefinitionURI** - the URI, OID or alternateID of the resource to check for existence (Type: ExternalURI).
- **context_{OPT}** - parameters that control the language, date, and other contextual variables (Type: ReadContext).

Return Type: `Boolean`

Exceptions

- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.

5.1.1.2 Operation: read

Retrieve the specified value set definition.

Input Parameters

- **valueSetDefinitionURI** - the URI, OID or alternateID of the resource to read (Type: `ExternalURI`).
- **queryControl_{OPT}** - parameters that control the ordering, timeout, and query behavior (Type: `QueryControl`).
- **context_{OPT}** - parameters that control the language, date, and other contextual variables (Type: `ReadContext`).

Return Type: `ValueSetDefinition`

Exceptions

- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The `format` is not supported by the service implementation.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.

5.1.1.3 Operation: existsDefinitionForValueSet

Determine if the value set definition for the specified value set exists on the service.

Input Parameters

- **valueSet** - the local name, about URI, OID or alternateID of a value set known to the service (Type: NameOrURI)
- **tag_{OPT}** - the version tag to use. If not specified, the CURRENT tag will be used. (Type: NameOrURI)
- **context_{OPT}** - parameters that control the language, date, and other contextual variables. (Type: ReadContext)

Return Type: Boolean

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **UnsupportedVersionTag** - The versionTag is not recognized by the service.
- **UnsupportedLanguage** - The referenceLanguage is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnsupportedContext** - One or more changeSetContext is not supported by the service.

5.1.1.4 Operation: readDefinitionForValueSet

Return the definition for the supplied value set and version tag.

Input Parameters

- **valueSet** - the local name, about URI, OID or alternateID of a value set known to the service (Type: NameOrURI)
- **tag_{OPT}** - the name or URI of a version tag. If not specified, the CURRENT tag will be used. (Type: NameOrURI)
- **context_{OPT}** - parameters that control the language, date, and other contextual variables. (Type: ReadContext)

Return Type: ValueSetDefinition

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **UnsupportedVersionTag** - The versionTag is not recognized by the service.
- **UnsupportedLanguage** - The referenceLanguage is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnsupportedContext** - One or more changeSetContext is not supported by the service.

5.2.1 Interface ValueSetDefinitionQueryService

Superclasses

Every instance of `ValidService` is also an instance of `BaseService`.

- #### 5.2.1.1 Operation: restrictToCodeSystem

Restrict the set of referenced definitions to those which reference the named code system.

Input Parameters

ValuesSetDefinitionsFactory, a `ValuesSetDefinitionsFactory` that resolves to a set of value set definitions

- 34 CTS2 - Value Set Services, v1.0

Return Type: ValueSetDefinitionDirectoryURI

Exceptions

- **UnknownCodeSystem** - The referenced CodeSystem is not recognized by the service.
- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.

5.2.1.2 Operation: resolve

Resolve a ValueSetDefinitionDirectoryURI to a directory summarizing the referenced definitions.

Input Parameters

- **directory** - a DirectoryURI that references a set of ValueSetDefinitions (Type: ValueSetDefinitionDirectoryURI).
- **queryControl_{OPT}** - parameters that control the ordering, timeout, and query behavior (Type: QueryControl).
- **context** - parameters that control the language, date, and other contextual variables (Type: ReadContext).

Return Type: ValueSetDefinitionDirectory

Exceptions

- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.
- **UnsupportedLanguage** - The referenceLanguage is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnsupportedContext** - One or more changeSetContext is not supported by the service.
- **UnsupportedLanguage** - The referenceLanguage is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnsupportedContext** - One or more changeSetContext is not supported by the service.

5.2.1.3 Operation: resolveAsList

Resolve a ValueSetDefinitionDirectoryURI to a list containing the referenced definitions.

Input Parameters

- **directory** - a DirectoryURI that references a set of ValueSetDefinitions (Type: ValueSetDefinitionDirectoryURI).

- **returnFilter**_{OPT} - a list of zero or more component references. If present, the returned list entries will contain only the required elements and attributes plus any elements or attributes named in the filter. If absent all elements and attributes will be returned. (Type: ComponentReferenceList).
- **queryControl**_{OPT} - parameters that control the ordering, timeout, and query behavior (Type: QueryControl).
- **context** - parameters that control the language, date, and other contextual variables (Type: ReadContext).

Return Type: ValueSetDefinitionList

Exceptions

- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.

5.3 Value Set History Service



Figure 5.3 - Value Set Definition History

The ValueSetDefinitionHistoryService allows the client to find the first change, last change, and the set of changes that occurred to a ValueSetDefinition in a specified time interval.

5.3.1 Interface ValueSetDefinitionHistoryService

A service that provides the change history for a value set definition.

Superclasses

- Every instance of ValueSetDefinitionHistoryService is also an instance of BaseHistoryService.

5.3.1.1 Operation: getEarliestChangeFor

Return the first known state of the specified ValueSetCatalogEntry.

Input Parameters

- **valueSetDefinition** - the URI of a value set definition (Type: ExternalURI).

Return Type: ValueSetDefinition

Exceptions

- **UnknownValueSetDefinition** - The valueSetDefinition URI isn't recognized by the service.

5.3.1.2 Operation: getLastChangeFor

Return the last change made to the specified ValueSetCatalogEntry.

Input Parameters

- **valueSetDefinition** - the URI of a value set definition (Type: ExternalURI).

Return Type: ValueSetDefinition

Exceptions

- **UnknownValueSetDefinition** - The valueSetDefinition URI isn't recognized by the service.

5.3.1.3 Operation: getChangeHistoryFor

Return the set of states and changes that occurred to the specified ValueSetCatalogEntry within the specified time period.

Input Parameters

- **valueSetDefinition** - the URI of a value set definition. (Type: ExternalURI)
- **fromDate_{OPT}** - the starting date. If not supplied, states beginning with the first known state will be returned. If later that toDate, the return order is earliest to latest, otherwise it is latest to earliest. (Type: DateAndTime)
- **toDate_{OPT}** - the ending date. If not supplied, states up to the last known state will be returned. If later that fromDate, the return order is latest to earliest, otherwise it is earliest to latest. (Type: DateAndTime)

Return Type: ValueSetDefinitionList

Exceptions

- **UnknownValueSetDefinition** - The valueSetDefinition URI isn't recognized by the service.

5.4 Value Set Definition Maintenance Services

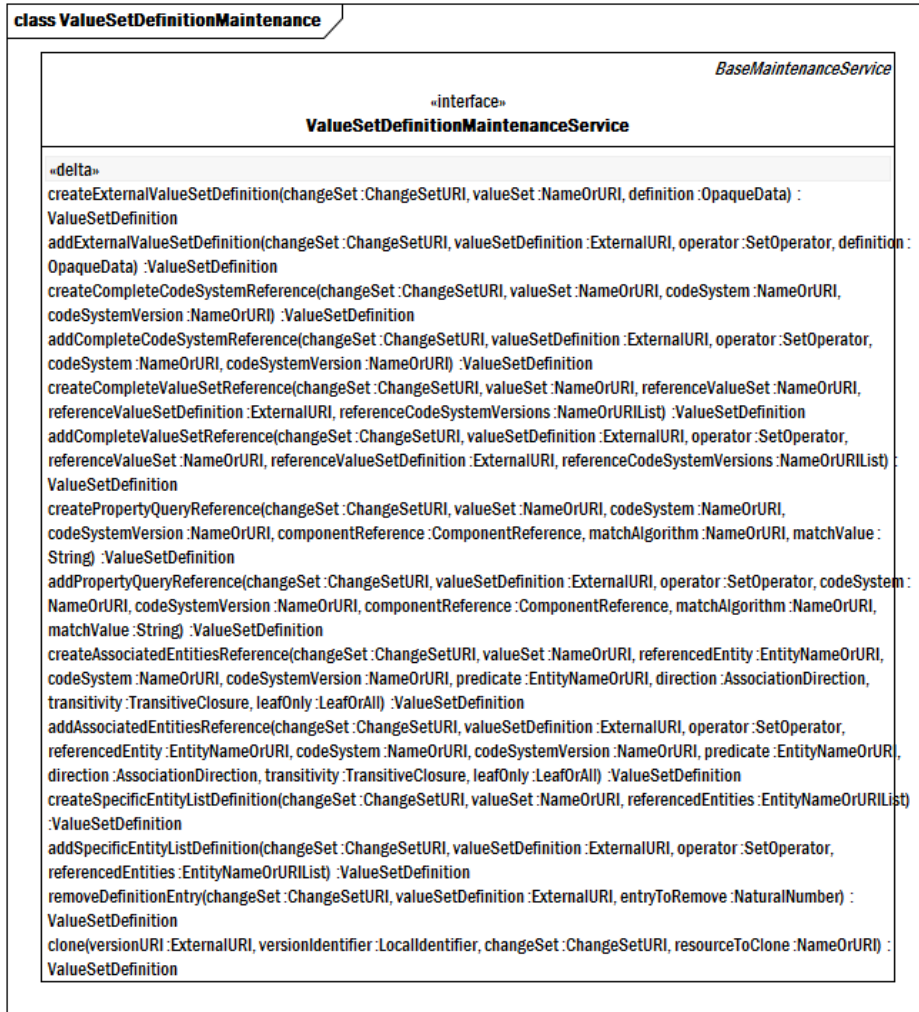


Figure 5.4 - Value Set Definition Maintenance

The ValueSetDefinitionMaintenanceService allows the creation, modification, and deletion of ValueSetDefinitions. The attributes and operations provided by the BaseMaintenanceService have been omitted from this diagram to reduce the complexity of the diagram.

5.4.1 Interface ValueSetDefinitionMaintenanceService

A service that allows the creation, update, and removal of value set definitions.

- Every instance of ValueSetDefinitionMaintenanceService is also an instance of BaseMaintenanceService.

5.4.1.1 Operation: createExternalValueSetDefinition

Create a new value set definition consisting of an `ExternalValueSetDefinition` with the operator set to `UNION`.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: `ChangeSetURI`).
- **valueSet** - the name or URI of the defined value set (Type: `NameOrURI`).
- **definition** - the external definition (Type: `OpaqueData`).

Return Type: `ValueSetDefinition`

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.

5.4.1.2 Operation: addExternalValueSetDefinition

Add an `ExternalValueSetDefinition` component to an existing `ValueSetDefinition`.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: `ChangeSetURI`).
- **valueSetDefinition** - the URI of an existing value set definition (Type: `ExternalURI`).
- **operator** - an indicator that determines whether the referenced set is to be added (`UNION`), removed (`SUBTRACT`), or anded (`INTERSECT`) with the current set of values (Type: `SetOperator`).
- **definition** - the external definition (Type: `OpaqueData`).

Return Type: `ValueSetDefinition`

Exceptions

- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.

5.4.1.3 Operation: createCompleteCodeSystemReference

Create a new value set definition consisting of a `CompleteCodeSystemReference` with the `operator` set to `UNION` referencing the supplied code system and optional version. A service implementation MAY choose to validate the code system URI and/or verify that the supplied version is valid for the code system. The implementation MAY, instead, assume the supplied URIs are valid and simply unknown to the service at this point in time.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: `ChangeSetURI`).
- **valueSet** - the name or URI of the value set to be defined (Type: `NameOrURI`).
- **codeSystem** - the name or URI of the referenced code system (Type: `NameOrURI`).
- **codeSystemVersion_{OPT}** - the optional name or URI of a version of the code system (Type: `NameOrURI`).

Return Type: ValueSetDefinition

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **UnknownCodeSystem** - The referenced `CodeSystem` is not recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **CodeSystemVersionMismatch** - The named code system version is not a version of the named code system.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not OPEN.

5.4.1.4 Operation: addCompleteCodeSystemReference

Add a `CompleteCodeSystemReference` to an existing `ValueSetDefinition`.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: `ChangeSetURI`).
- **valueSetDefinition** - The name or URI of an existing value set definition (Type: `ExternalURI`).
- **operator** - An indicator that determines whether the referenced set is to be added (`UNION`), removed (`SUBTRACT`) or anded (`INTERSECT`) with the current set of values (Type: `SetOperator`).
- **codeSystem** - The name or URI of the referenced code system (Type: `NameOrURI`).
- **codeSystemVersion_{OPT}** - The optional name or URI of a version of the code system (Type: `NameOrURI`).

Return Type: ValueSetDefinition

Exceptions

- **UnknownValueSetDefinition** - The valueSetDefinition URI isn't recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **UnknownCodeSystem** - The referenced CodeSystem is not recognized by the service.
- **UnknownCodeSystemVersion** - The referenced CodeSystemVersion is not recognized by the service.
- **CodeSystemVersionMismatch** - The named code system version is not a version of the named code system.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.

5.4.1.5 Operation: createCompleteValueSetReference

Create a new value set definition consisting of a CompleteValueSetReference with the operator set to UNION referencing the supplied code system and optional version. A service implementation MAY choose to validate the value set URI and/or verify that the supplied version is valid for the value set. The implementation MAY, instead, assume the supplied URIs are valid and simply unknown to the service at this point in time.

The service implementation MAY choose to validate the supplied code system versions and/or validate whether the versions are referenced in the supplied (optional) value set definition.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: ChangeSetURI).
- **valueSet** - the name or URI of the defined value set (Type: NameOrURI).
- **referenceValueSet** - the name or URI of the referenced value set (Type: NameOrURI).
- **referenceValueSetDefinition_{OPT}** - the optional URI of a specific definition of the value set (Type: ExternalURI).
- **referenceCodeSystemVersions_{OPT}** - a optional list of the names or URIs code system versions to use in resolution (Type: NameOrURIList).

Return Type: ValueSetDefinition

Exceptions

- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **UnknownCodeSystemVersion** - The referenced CodeSystemVersion is not recognized by the service.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.

- **ValueSetDefinitionMismatch** - The named value set definition was not a definition for the named value set.

5.4.1.6 Operation: addCompleteValueSetReference

Add a `CompleteValueSetReference` component to an existing `ValueSetDefinition`.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: `ChangeSetURI`).
- **valueSetDefinition** - the URI of an existing value set definition (Type: `ExternalURI`).
- **operator** - an indicator that determines whether the referenced set is to be added (UNION), removed (SUBTRACT), or anded (INTERSECT) with the current set of values (Type: `SetOperator`).
- **referenceValueSet** - the name or URI of the referenced value set (Type: `NameOrURI`).
- **referenceValueSetDefinitionOP** - The optional name or URI of a specific definition of the value set (Type: `DocumentURI`).
- **referenceCodeSystemVersionsOP** - an optional list of the names or URIs code system versions to use in resolution. (Type: `NameOrURIList`).

Return Type: ValueSetDefinition

Exceptions

- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not OPEN.
- **ValueSetDefinitionMismatch** - The named value set definition was not a definition for the named value set.

5.4.1.7 Operation: createPropertyQueryReference

Create a new value set definition consisting of a `PropertyQueryReference` with the operator set to UNION.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: `ChangeSetURI`).
- **valueSet** - the name or URI of the value set being defined (Type: `NameOrURI`).
- **codeSystem** - the name or URI of the code system that contains the property assertions (Type: `NameOrURI`).
- **codeSystemVersion** - the optional name or URI of a specific code system version (Type: `NameOrURI`).

- **componentReference** - A reference to a CTS2 model element. ComponentReference may reference a model attribute, a Property or a special element such as match strength (Type: ComponentReference).
- **matchAlgorithm** - the name or URI of a match algorithm (Type: NameOrURI).
- **matchValue** - the value to match if required by the algorithm (Type: String).

Return Type: ValueSetDefinition

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnknownEntity** - The EntityNameOrURI is not known to the service.
- **UnsupportedMatchAlgorithm** - The matchAlgorithm is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.

5.4.1.8 Operation: addPropertyQueryReference

Add a PropertyQueryReference to the end of an existing ValueSetDefinition.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: ChangeSetURI).
- **valueSetDefinition** - the URI of the existing value set definition (Type: ExternalURI).
- **operator** - an indicator that determines whether the referenced set is to be added, removed, or anded (Type: SetOperator).
- **codeSystem** - the name or URI of the code system that contains the property assertions (Type: NameOrURI).
- **codeSystemVersion** - the optional name or URI of a specific code system version (Type: NameOrURI).
- **componentReference** - A reference to a CTS2 model element. ComponentReference may reference a model attribute, a Property or a special element such as match strength (Type: ComponentReference).
- **matchAlgorithm** - the name or URI of a match algorithm (Type: NameOrURI).

- **matchValue** - the value to match if required by the algorithm (Type: String).

Return Type: ValueSetDefinition

Exceptions

- **UnknownValueSetDefinition** - The valueSetDefinition URI isn't recognized by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnknownEntity** - The EntityNameOrURI is not known to the service.
- **UnsupportedMatchAlgorithm** - The matchAlgorithm is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.

5.4.1.9 Operation: createAssociatedEntitiesReference

Create a new value set definition consisting of an AssociatedEntitiesReference with the operator set to UNION referencing the supplied set of entities.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: ChangeSetURI).
- **valueSet** - the name or URI of the value set being defined (Type: NameOrURI).
- **referencedEntity** - the namespace/name or URI of the reference entity for the association (Type: EntityNameOrURI).
- **codeSystem** - the name or URI of the code system that contains the associations to be referenced (Type: NameOrURI).
- **codeSystemVersion** - the optional name or URI of the specific code system version making the assertions. (Type: NameOrURI).
- **predicate** - the namespace/name or URI of the predicate of the association (Type: EntityNameOrURI).
- **direction** - the association direction (SOURCE_TO_TARGET or TARGET_TO_SOURCE) (Type: AssociationDirection).
- **transitivity**_{OPT} - whether the closure of the association is included. If omitted this defaults to DIRECTLY_ASSOCIATED (Type: TransitiveClosure).

- **leafOnly_{OPT}** - the LEAF_ONLY or ALL_INTERMEDIATE_NODES indicator. If omitted, ALL_INTERMEDIATE_NODES is assumed (Type: LeafOrAll).

Return Type: ValueSetDefinition

Exceptions

- **UnknownCodeSystem** - The referenced CodeSystem is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **CodeSystemVersionMismatch** - The named code system version is not a version of the named code system.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **UnknownCodeSystemVersion** - The referenced CodeSystemVersion is not recognized by the service.
- **UnknownEntity** - The EntityNameOrURI is not known to the service.

5.4.1.10 Operation: addAssociatedEntitiesReference

Add an AssociatedEntitiesReference component to an existing ValueSetDefinition.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - The URI of an OPEN change set that will contain this change. (Type: ChangeSetURI)
- **valueSetDefinition** - the URI of an existing ValueSetDefinition. (Type: ExternalURI)
- **operator** - an indicator that determines whether the referenced set is to be added (UNION), removed (SUBTRACT), or anded (INTERSECT) with the current set of values. (Type: SetOperator)
- **referencedEntity** - the namespace/name or URI of the reference entity for the association. (Type: EntityNameOrURI)
- **codeSystem** - the name or URI of the code system that contains the associations to be referenced. (Type: NameOrURI)
- **codeSystemVersion** - the optional name or URI of the specific code system version making the assertions. (Type: NameOrURI)
- **predicate** - the namespace/name or URI of the predicate of the association. (Type: EntityNameOrURI)
- **direction** - the association direction (SOURCE_TO_TARGET or TARGET_TO_SOURCE). (Type: AssociationDirection)
- **transitivity_{OPT}** - whether the closure of the association is included. If omitted this defaults to DIRECTLY_ASSOCIATED. (Type: TransitiveClosure)

- **leafOnly_{OPT}** - the LEAF_ONLY or ALL_INTERMEDIATE_NODES indicator. If omitted, ALL_INTERMEDIATE_NODES is assumed. (Type: LeafOrAll)

Return Type: ValueSetDefinition

Exceptions

- **UnknownCodeSystem** - The referenced CodeSystem is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnknownValueSetDefinition** - The valueSetDefinition URI isn't recognized by the service.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **CodeSystemVersionMismatch** - The named code system version is not a version of the named code system.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **UnknownCodeSystemVersion** - The referenced CodeSystemVersion is not recognized by the service.
- **UnknownEntity** - The EntityNameOrURI is not known to the service.

5.4.1.11 Operation: createSpecificEntityListDefinition

Create a new value set definition consisting of a SpecificEntityList with the operator set to UNION referencing the supplied set of entities.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change. (Type: ChangeSetURI)
- **valueSet** - the name or URI of the value set being defined. (Type: NameOrURI)
- **referencedEntities** - the namespace/names or URIs of the referenced entities. (Type: EntityNameOrURIList)

Return Type: ValueSetDefinition

Exceptions

- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnknownEntity** - The EntityNameOrURI is not known to the service.

5.4.1.12 Operation: addSpecificEntityListDefinition

Add a SpecificEntityList component to an existing ValueSetDefinition.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set that will contain this change (Type: `ChangeSetURI`).
- **valueSetDefinition** - the URI of an existing `ValueSetDefinition` (Type: `ExternalURI`).
- **operator** - an indicator that determines whether the referenced set is to be added (`UNION`), removed (`SUBTRACT`), or anded (`INTERSECT`) with the current set of values (Type: `SetOperator`).
- **referencedEntities** - the namespace/names or URIs of the referenced entities (Type: `EntityNameOrURIList`).

Return Type: `ValueSetDefinition`

Exceptions

- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnknownEntity** - The `EntityNameOrURI` is not known to the service.

5.4.1.13 Operation: `removeDefinitionEntry`

Remove the specified definition entry from the named value set.

This operation may alter the state of the service.

Input Parameters

- **changeSet** - the URI of an OPEN change set to contain this change (Type: `ChangeSetURI`).
- **valueSetDefinition** - the URI of an existing value set definition (Type: `DocumentURI`).
- **entryToRemove** - the one-based index of the particular entry to remove (Type: `NaturalNumber`).

Return Type: `ValueSetDefinition`

Exceptions

- **InvalidIndexNumber** - The supplied number does not reference an existing `ValueSetDefinitionEntry` or the operation would remove the last `ValueSetDefinitionEntry` from a definition.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **ResourceIsNotOpen** - The target resource version description has been finalized and cannot be updated.

5.4.1.14 Operation: clone

Create an copy of the existing `ValueSetDefinition` and the referenced `Associations` and `ValueSetDefinitionEntries`, assigning a new version identifier and about URI. The cloned resource will be marked as `OPEN`, allowing incremental changes to occur.

This operation may alter the state of the service.

Input Parameters

- **versionURI_{OPT}** - the about URI of the new cloned resource. If absent, it is assumed that the service will generate the new URI. (Type: `ExternalURI`)
- **versionIdentifier_{OPT}** - the local identifier of the new resource version. If absent, the system is expected to assign the identifier. (Type: `LocalIdentifier`)
- **changeSet** - the URI of an `OPEN` change set where the clone operation will be recorded (Type: `ChangeSetURI`)
- **resourceToClone** - the local identifier or about URI of the value set definition to be cloned (Type: `NameOrURI`)

Return Type: `ValueSetDefinition`

Exceptions

- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.

6 Value Set Definition Resolution Services

6.1 Value Set Resolution Services

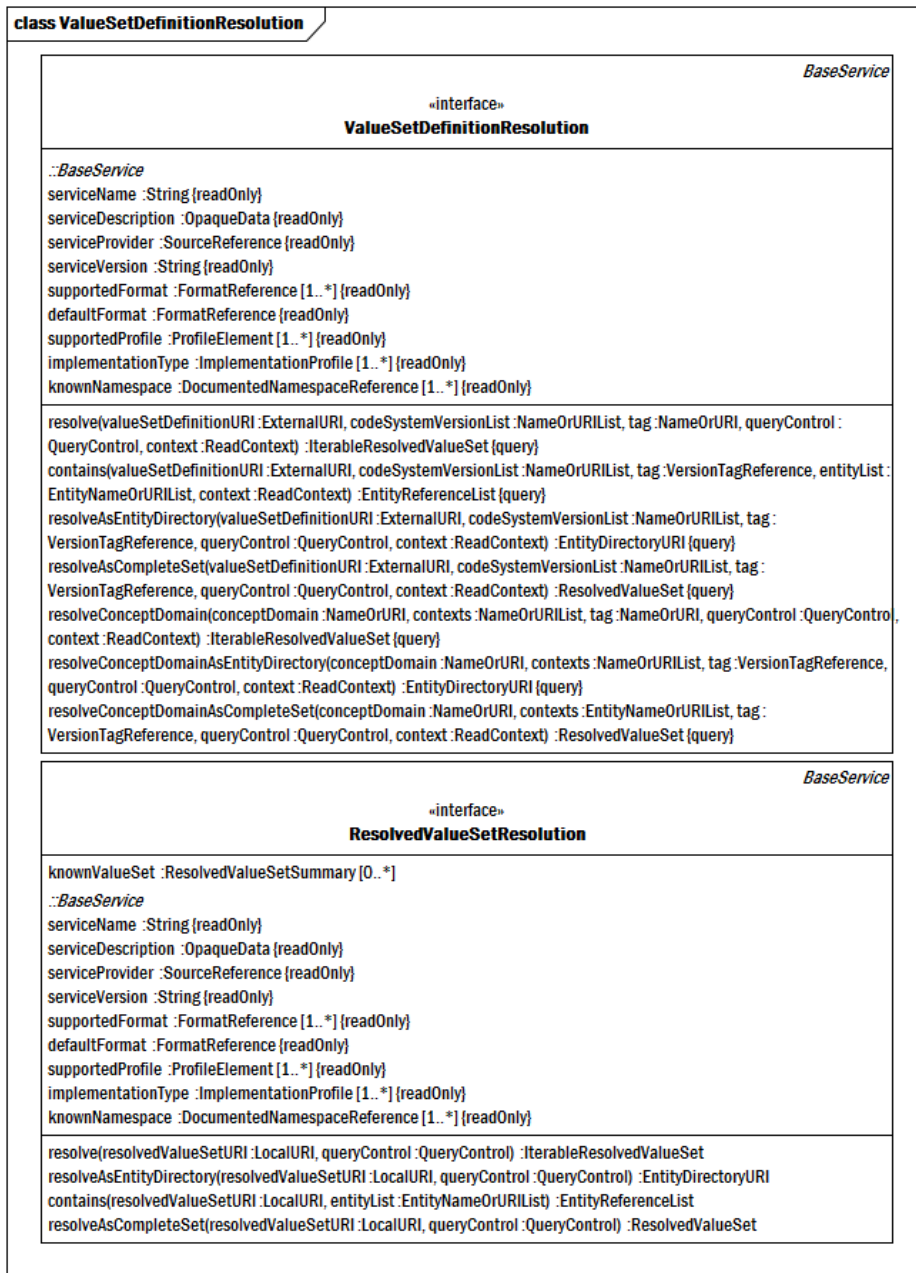


Figure 6.1 - Value Set Resolution Service

6.1.1 Interface ValueSetDefinitionResolution

The `ValueSetResolutionService` provides the ability to resolve value set definitions against code system versions - versions that can either be supplied directly or via version tags. The resolution of a value set can be returned as an “iterable directory” an entity directory that can be used in other entity directory functions (useful only if the service supports the ENTITY profile), or as a resolved value set. Value sets can also be tested for the presence or absence of entities.

Superclasses

- Every instance of `ValueSetDefinitionResolution` is also an instance of `BaseService`.

6.1.1.1 Operation: resolve

Resolve the value set definition against the supplied code system versions and/or version tags.

Input Parameters

- **valueSetDefinitionURI** - the URI of a value set definition (Type: `DocumentURI`).
- **codeSystemVersionList**_{OPT} - a list of the names or URIs of code system versions (Type: `NameOrURIList`).
- **tag**_{OPT} - the name or URI of a version tag. If not present, the CURRENT tag is used (Type: `NameOrURI`).
- **queryControl**_{OPT} - parameters that control the ordering, timeout, and query behavior (Type: `QueryControl`).
- **context**_{OPT} - parameters that control the language, date, and other contextual variables (Type: `ReadContext`).

Return Type: IterableResolvedValueSet

Exceptions

- **UnsupportedVersionTag** - The `versionTag` is not recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not OPEN.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The `format` is not supported by the service implementation.

- **QueryTimeout** - The `timeLimit` was exceeded by the service.
- **InvalidDirectoryURI** - The supplied `directory` URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied `directory` URI is not the one required by the service invocation.

6.1.1.2 Operation: **resolveAsEntityDirectory**

Resolve the value set definition using the supplied code system versions and/or version tag and return a `DirectoryURI` that resolves to the set of entities in the resolved set.

Input Parameters

- **valueSetDefinitionURI** - the URI of a value set (Type: `DocumentURI`).
- **codeSystemVersionList_{OPT}** - a list of code system version names and/or URIs (Type: `NameOrURIList`).
- **tag_{OPT}** - the name or URI of a version tag (Type: `VersionTagReference`).
- **queryControl_{OPT}** - parameters that control the ordering, timeout, and query behavior (Type: `QueryControl`).
- **context_{OPT}** - parameters that control the language, date, and other contextual variables (Type: `ReadContext`).

Return Type: **EntityDirectoryURI**

Exceptions

- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The `format` is not supported by the service implementation.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.

6.1.1.3 Operation: contains

Intersect the supplied entity list with what would result were the value set, when resolved against the supplied code system version and/or version tag and return those that are in common.

Input Parameters

- **valueSetDefinitionURI** - the URI of a value set definition (Type: DocumentURI).
- **codeSystemVersionList_{OPT}** - the name and/or URIs of a set of code system versions to resolve against (Type: NameOrURIList).
- **tag_{OPT}** - the name or URI of a version tag. Defaults to CURRENT. (Type: VersionTagReference)
- **entityList** - the namespace/name or URIs of the entities to test for membership (Type: EntityNameOrURIList).
- **context_{OPT}** - parameters that control the language, date, active status, and other contextual variables (Type: ReadContext).

Return Type: EntityReferenceList

Exceptions

- **UnknownValueSetDefinition** - The valueSetDefinition URI isn't recognized by the service.
- **UnknownCodeSystemVersion** - The referenced CodeSystemVersion is not recognized by the service.
- **UnsupportedLanguage** - The referenceLanguage is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN.
- **UnsupportedContext** - One or more changeSetContext is not supported by the service.

6.1.1.4 Operation: resolveAsCompleteSet

Resolve the value set definition against the supplied code system versions and/or tags and return a ResolvedValueSet that can be transferred and loaded into other systems.

Input Parameters

- **valueSetDefinitionURI** - the URI of a value set definition. (Type: DirectoryURI)
- **codeSystemVersionList_{OPT}** - the name or URI of the code system versions to resolve against. (Type: NameOrURIList)
- **tag_{OPT}** - the name or URI of a version tag. If not supplied, defaults to CURRENT. (Type: VersionTagReference)
- **queryControl_{OPT}** - parameters that control ordering, timeout, and query behavior (Type: QueryControl).
- **context_{OPT}** - parameters that control the language, date, and other contextual variables (Type: ReadContext).

Return Type: `ResolvedValueSet`

Exceptions

- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **UnsupportedVersionTag** - The `versionTag` is not recognized by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The `format` is not supported by the service implementation.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.

6.1.1.5 Operation: resolveConceptDomain

Resolve the supplied concept domain using the supplied contexts and version tag as a resolved value set.

Input Parameters

- **conceptDomain** - the URI of a the conceptDomain. (Type: `NameOrURI`)
- **contexts** - the name(s) or URI(s) of a set of realms or contexts (Type: `NameOrURIList`)
- **tag_{OPT}** - the name or URI of a version tag. If not supplied, defaults to `CURRENT`. (Type: `NameOrURI`)
- **queryControl_{OPT}** - parameters that control ordering, timeout, and query behavior (Type: `QueryControl`).
- **context_{OPT}** - parameters that control the language, date, and other contextual variables (Type: `ReadContext`).

Return Type: `IterableResolvedValueSet`

Exceptions

- **UnsupportedVersionTag** - The `versionTag` is not recognized by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.

- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.

6.1.1.6 Operation: `resolveConceptDomainAsCompleteSet`

Resolve the supplied concept domain using the supplied contexts and version tag as a complete resolved value set that can be transferred to other systems

Input Parameters

- **conceptDomain** - the URI of a the conceptDomain. (Type: `NameOrURI`)
- **contexts** - the name(s) or URI(s) of a set of realms or contexts (Type: `NameOrURIList`)
- **tag_{OPT}** - the name or URI of a version tag. If not supplied, defaults to `CURRENT`. (Type: `NameOrURI`)
- **queryControl_{OPT}** - parameters that control ordering, timeout, and query behavior (Type: `QueryControl`).
- **context_{OPT}** - parameters that control the language, date, and other contextual variables (Type: `ReadContext`).

Return Type: `ResolvedValueSet`

Exceptions

- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service
- **UnsupportedVersionTag** - The `versionTag` is not recognized by the service
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The `format` is not supported by the service implementation
- **QueryTimeout** - The `timeLimit` was exceeded by the service.

6.1.1.7 Operation: **resolveConceptDomainAsEntityDirectory**

Resolve the supplied concept domain using the supplied contexts and version tag as an entity directory.

Input Parameters

- **conceptDomain** - the URI of a the conceptDomain. (Type: NameOrURI)
- **contexts** - the name(s) or URI(s) of a set of realms or contexts (Type: NameOrURIList)
- **tag_{OPT}** - the name or URI of a version tag. If not supplied, defaults to CURRENT. (Type: NameOrURI)
- **queryControl_{OPT}** - parameters that control ordering, timeout, and query behavior (Type: QueryControl).
- **context_{OPT}** - parameters that control the language, date, and other contextual variables (Type: ReadContext).

Return Type: EntityDirectoryURI

Exceptions

- **UnknownValueSetDefinition** - The valueSetDefinition URI isn't recognized by the service.
- **UnknownCodeSystemVersion** - The referenced CodeSystemVersion is not recognized by the service
- **UnsupportedMatchAlgorithm** - The matchAlgorithm is not supported by the service
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The format is not supported by the service implementation
- **QueryTimeout** - The timeLimit was exceeded by the service.
- **UnsupportedLanguage** - The referenceLanguage is not supported by the service
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The changeSetContext is recognized by the service, but its state is not OPEN .

UnsupportedContext - One or more changeSetContext is not supported by the service

7 Resolved Value Set Services

7.1 Resolved Value Set Resolution Service

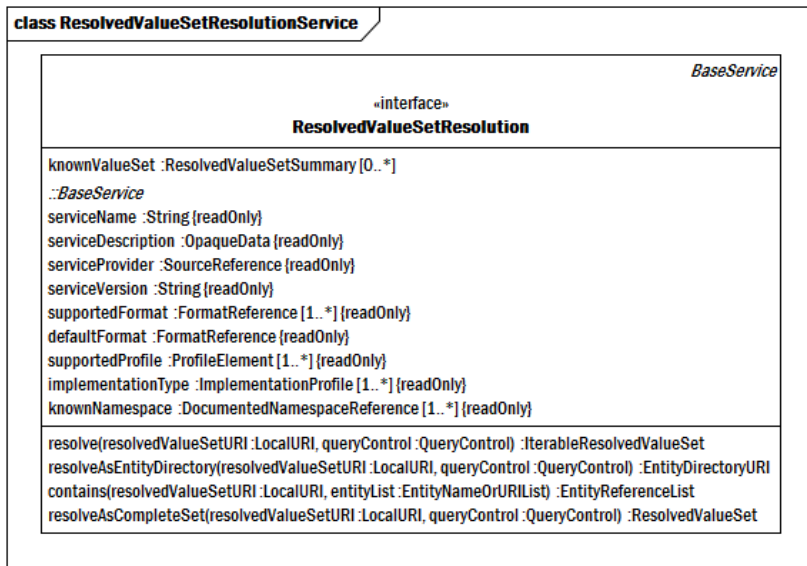


Figure 7.1 - Value Set Resolution Service

7.1.1 Interface ResolvedValueSetResolution

The **ResolvedValueSetResolution** service provides the ability query and return previously resolved value sets that have been uploaded to the local service. The resolution can be returned as an iterable directory, an entity directory that can be used in other entity directory functions (useful only if the service supports the ENTITY profile), or as a full resolved value set.

Superclasses:

- Every instance of **ResolvedValueSetResolution** is also an instance of **BaseService**.

Attributes:

- **knownValueSet** - The set of resolved value sets that are known to the service.

7.1.1.1 Operation: resolve

Return an iterator for the entire resolved set.

This operation may alter the state of the service.

Input Parameters:

- **resolvedValueSetURI** - The URI of a resolved value set. (Type: LocalURI)
- **queryControl** - parameters that control the ordering, timeout, and query behavior. (Type: QueryControl)

Return Type: IterableResolvedValueSet

Exceptions:

- **UnsupportedVersionTag** - The `versionTag` is not recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The format is not supported by the service implementation.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.
- **InvalidDirectoryURI** - The supplied `directory` URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied `directory` URI is not the one required by the service invocation.

7.1.1.2 Operation: resolveAsEntityDirectory

Return a `DirectoryURI` that resolves to the set of entities in the resolved set.

This operation may alter the state of the service.

Input Parameters:

- **resolvedValueSetURI** - the URI of a resolved value set. (Type: LocalURI)
- **queryControl** - parameters that control the ordering, timeout, and query behavior. (Type: QueryControl)

Return Type: `EntityDirectoryURI`

Exceptions:

- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The format is not supported by the service implementation.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service

7.1.1.3 Operation: `contains`

Intersect the supplied entity list with what the entities in the resolved set and return those that are in common.

This operation may alter the state of the service.

Input Parameters:

- **resolvedValueSetURI** - the URI of a resolved value set. (Type: `LocalURI`)
- **entityList** - the namespace/name or URIs of the entities to test for membership. (Type: `EntityNameOrURIList`)

Return Type: `EntityReferenceList`

Exceptions:

- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.

7.1.1.4 Operation: resolveAsCompleteSet

Return the `ResolvedValueSet` in a form that can be transferred and loaded into other systems.

This operation may alter the state of the service.

Input Parameters:

- **resolvedValueSetURI** - the URI of a value set definition. (Type: `LocalURI`)
- **queryControl** - parameters that control ordering, timeout, and query behavior. (Type: `QueryControl`)

Return Type: `ResolvedValueSet`

Exceptions:

- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **UnsupportedVersionTag** - The `versionTag` is not recognized by the service.
- **UnsupportedLanguage** - The `referenceLanguage` is not supported by the service.
- **UnknownChangeSet** - The change set specified could either not be read or located by the service.
- **ChangeSetIsNotOpen** - The `changeSetContext` is recognized by the service, but its state is not `OPEN`.
- **UnsupportedContext** - One or more `changeSetContext` is not supported by the service.
- **UnsupportedMatchAlgorithm** - The `matchAlgorithm` is not supported by the service.
- **UnsupportedPredicate** - The predicate name or URI is not recognized by the service.
- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnsupportedModelAttribute** - The name or URI of a CTS2 model attribute is not recognized and/or supported by the service implementation.
- **MatchValueFormatException** - The match value does not meet the format or syntax required or the supplied match algorithm.
- **UnsupportedFormat** - The `format` is not supported by the service implementation.
- **QueryTimeout** - The `timeLimit` was exceeded by the service.

7.2 ResolvedValueSetLoader

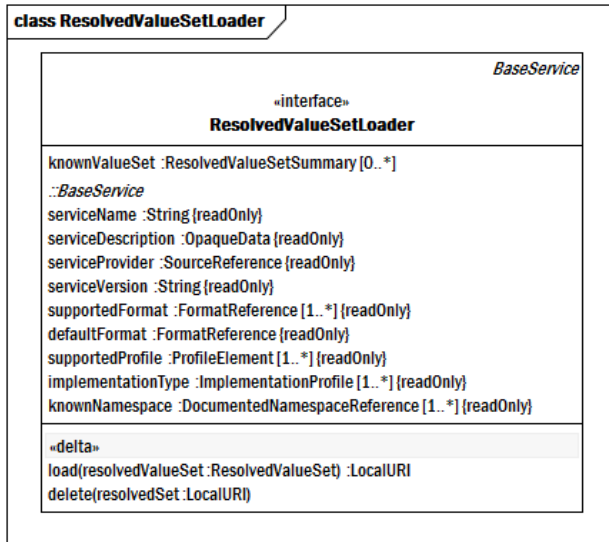


Figure 7.2 - Resolved Value Set Loader

7.2.1 Interface ResolvedValueSetLoader

The `ResolvedValueSetLoader` provides the ability to load and test resolved value sets without necessarily having the supporting code system versions available. It provides the ability to load and delete resolved value sets and to determine whether a resolved set contains one or more entities. It also provides the ability to access the contents of a resolved set, albeit without iterators.

Superclasses

- Every instance of `ResolvedValueSetLoader` is also an instance of `BaseService`.

Attributes

- **knownValueSet** - the resolved value sets that have been loaded into the service.

7.2.1.1 Operation: load

Load the resolved value set into the local resolution list and return a URI that references it.

This operation may alter the state of the service.

Input Parameters

- **resolvedValueSet** - the resolved value set to load (Type: `ResolvedValueSet`).

Return Type: LocalURI

7.2.1.2 Operation: delete

Remove the referenced set from the service.

This operation may alter the state of the service.

Input Parameters

- **resolvedSet** - the resolved set to remove (Type: LocalURI).

Return Type: void

Exceptions

- **UnrecognizedResolvedValueSet** - the LocalURI is not the name of a valid resolved value set.

7.3 Resolved Value Set Query Service

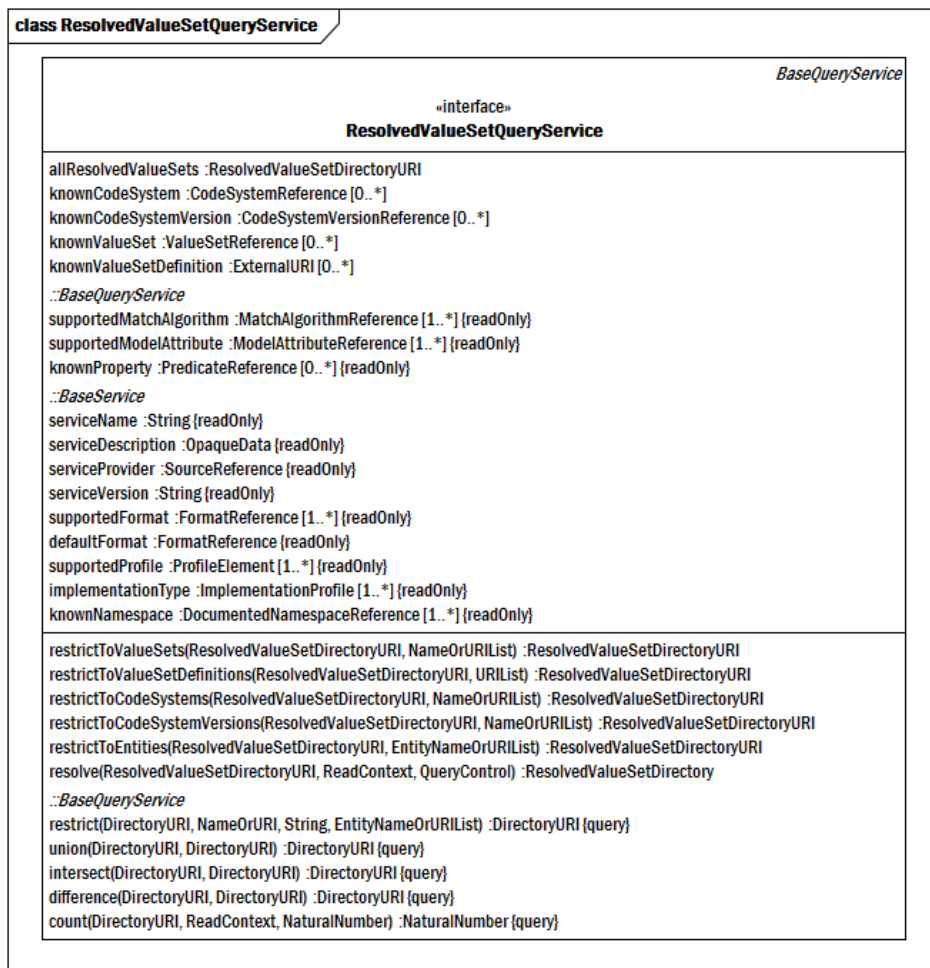


Figure 7.3 - Resolved Value Set Resolution Service

7.3.1 Interface ResolvedValueSetQueryService

A query service that allows the querying of resolved value sets. This service is used to discover resolved value sets for particular value sets, value set definitions, code systems, code system versions, or specific entities.

Superclasses:

- Every instance of `ResolvedValueSetQueryService` is also an instance of `BaseQueryService`.

Attributes:

- **allResolvedValueSets** - All resolved value sets that are known to the service.
- **knownCodeSystem** - The set of code systems that are known to this service. At a minimum, this list must contain all code systems that are used in resolved value sets.
- **knownCodeSystemVersion** - The set of code system versions that are known to this service. At a minimum, this list must contain all versions that are used in resolved value sets.
- **knownValueSet** -
- **knownValueSetDefinition** - The set of value set definitions that are known to the service. At a minimum this must contain all definitions that are used in the set of resolved value sets.

7.3.1.1 Operation: restrictToValueSets

Return the set of resolved value sets that were resolved against the supplied value set.

This operation may alter the state of the service.

Input Parameters:

- **directory** - a `DirectoryURI` that resolves to a set of `ResolvedValueSetSummarie`. (Type: `ResolvedValueSetDirectoryURI`)
- **valueSet** - Restrict the set of resolved value sets to those that were resolved partially or completely against the supplied value set URI. (Type: `NameOrURIList`)

Return Type: ResolvedValueSetDirectoryURI

Exceptions:

- **UnknownValueSet** - The value set name or URI is not recognized by the service.
- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.

7.3.1.2 Operation: restrictToValueSetDefinitions

Return the set of resolved value sets that resolved against the supplied definition(s).

This operation may alter the state of the service.

Input Parameters:

- **directory** - a `DirectoryURI` that resolves to a set of `ResolvedValueSetSummaries`. (Type: `ResolvedValueSetDirectoryURI`)
- **definitions** - The URIs of the value set definition to resolve against (Type: `URIList`)

Return Type: `ResolvedValueSetDirectoryURI`

Exceptions:

- **UnknownValueSetDefinition** - The `valueSetDefinition` URI isn't recognized by the service.
- **InvalidDirectoryURI** - The supplied directory URI is not valid
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.

7.3.1.3 Operation: `restrictToCodeSystems`

Return the set of resolved value sets that resolved against the supplied code system(s).

This operation may alter the state of the service.

Input Parameters:

- **directory** - a `DirectoryURI` that resolves to a set of `ResolvedValueSetSummaries`. (Type: `ResolvedValueSetDirectoryURI`)
- **codeSystems** - The set of code systems used as the filter. (Type: `NameOrURIList`)

Return Type: `ResolvedValueSetDirectoryURI`

Exceptions:

- **UnknownCodeSystem** - The referenced `CodeSystem` is not recognized by the service.
- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.

7.3.1.4 Operation: `restrictToCodeSystemVersions`

Return the set of resolved value sets that resolved against the supplied code system version(s).

This operation may alter the state of the service.

Input Parameters:

- **directory** - a `DirectoryURI` that resolves to a set of `ResolvedValueSetSummaries`. (Type: `ResolvedValueSetDirectoryURI`)
- **codeSystemVersions** - The set of code system versions used as the filter. (Type: `NameOrURIList`)

Return Type: ResolvedValueSetDirectoryURI

Exceptions:

- **UnknownCodeSystemVersion** - The referenced `CodeSystemVersion` is not recognized by the service.
- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.

7.3.1.5 Operation: restrictToEntities

Restrict the list to those whose resolution contains one or more of the supplied entities.

This operation may alter the state of the service.

Input Parameters:

- **directory** - a `DirectoryURI` that resolves to a set of `ResolvedValueSetSummaries`. (Type: `ResolvedValueSetDirectoryURI`)
- **list** - The set of entities used to constraint he directory. (Type: `EntityNameOrURIList`)

Return Type: ResolvedValueSetDirectoryURI

Exceptions:

- **UnsupportedNamespaceName** - The supplied namespace name is not one that is known to the service.
- **UnknownEntity** - The `EntityNameOrURI` is not known to the service
- **InvalidDirectoryURI** - The supplied directory URI is not valid.
- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.

7.3.1.6 Operation: resolve

Resolve the directory URI to the corresponding directory.

This operation may alter the state of the service.

Input Parameters:

- **directory** - a `DirectoryURI` that resolves to a set of `ResolvedValueSetSummaries`. (Type: `ResolvedValueSetDirectoryURI`)
- **context**_{OPT} - parameters that control the ordering, timeout, and query behavior. (Type: `ReadContext`)
- **queryControl**_{OPT} - parameters that control the ordering, timeout, and query behavior. (Type: `QueryControl`)

Return Type: ResolvedValueSetDirectory

Exceptions:

- **InvalidDirectoryURI** - The supplied directory URI is not valid.

- **InvalidDirectoryType** - The type represented by the supplied directory URI is not the one required by the service invocation.