# org.omg.orderservice.catalog

Package in package 'OrderCatalog'

org.omg.orderservice.catalog

Version 1.0 Phase 1.0 Proposed

lconstab created on 10/24/2016. Last modified 5/8/2017

## Order Catalog

Package in package 'org.omg.orderservice.catalog'

This is a high level description of the Order Catalog Management data structures. The detailed implementation of the normative description of the structures is in the machine readable files associated with this specification

Overview of Data Model

The Order Catalog data model is composed of data structures needed to support both the operations performed on a catalog of products and services and the entities managed by such catalogs. These structures are divided into three categories:

1. The Core Data Model defines the foundational structures used to represent a catalog and the entries contained therein.
2. The Support Data Structures package is composed of supporting data structures such as type enumerations, catalog entity metadata, and operation configuration options.
3. Common Model Elements (described in Section 14) contains data structures that are shared by both the Order Catalog Service and the Order Management Service.

Order Catalog

Version 1.0 Phase 1.0 Proposed

lconstab created on 10/27/2016. Last modified 6/6/2017

### Type Definitions diagram

Class diagram in package 'Order Catalog'

The Type Definition diagram introduces the data type metamodel for the declarative representation of CatalogItem detail models.

Type Definitions

Version 1.0

lconstab created on 8/25/2017. Last modified 8/25/2017



Type Definitions

### ContainerType

Class in package 'Classes'

An abstract collection type specifying the semantics of the collection and the type of its elements. The specification of the type of the elements of the collection are left to subclasses.

ContainerType

Version 1.0 Phase 1.0 Proposed

Claude Nanjo created on 2/11/2017. Last modified 8/25/2017

Extends TypeDefinition

| ATTRIBUTES |
| --- |
| ContainerType : ContainerTypeEnum Public  The type of the container. Restricted to the values for the ContainerTypeEnum.  [ Is static False. Containment is Not Specified. ] |

### SimpleType

Class in package 'Classes'

A simple type definition that specifies the actual type as a string.

| ATTRIBUTES |
| --- |
| TypeName : String Public  The name of the type.  [ Is static False. Containment is Not Specified. ] |

### SimpleTypeContainer

Class in package 'Classes'

A typed collection such as the definition of a set of String types.

| ATTRIBUTES |
| --- |
| elementType : String Public  The element type such as Collection<elementType>  [ Is static False. Containment is Not Specified. ] |

### TemplateReferenceType

Class in package 'Classes'

This is a type that contains a reference to a template.

| ATTRIBUTES |
| --- |
| Reference : Identifier Public  The reference to an orderable definition.  [ Is static False. Containment is Not Specified. ] |

### TemplateReferenceTypeContainer

Class in package 'Classes'

A collection of TemplateComponent references.

| ATTRIBUTES |
| --- |
| Reference : Identifier Public  The reference to an orderable definition.  [ Is static False. Containment is Not Specified. ] |

### TypeDefinition

Class in package 'Classes'

Parent metamodel class for type definitions. The current set of types supported for the representation of orderable item detail models are SimpleType, ContainerType, and TemplateReferenceType.

### Order Catalog diagram

Class diagram in package 'Order Catalog'

The following model depicts the platform independent model for an order service catalog.

At its core, an Order Service Catalog consists of entries represented by specializations of the AbstractEntry class. Entries can be grouped into CatalogEntryGroups which, themselves can contain other entry groups in recursive fashion. An EntryGroup is often represented as a section in a catalog. A CatalogEntry may reside within an EntryGroup, or at the top level position in a catalog. CatalogEntryReference allows an entry to be categorized in any number of EntryGroups.

A CatalogEntry is a wrapper that contextualizes a CatalogItem within a specific catalog. CatalogItems are typically definitional clinical artifacts (models not tied to a specific patient) such as medication products, laboratory tests, procedures, or knowledge artifacts such as order sets.

Catalog entries can be related to other catalog entries using the CatalogEntryRelationship association. The type of relationship is specified by the relationship type concept. Such associations are defined by the catalog itself (as opposed to CatalogItemRelationships which are relationships defined among CatalogItems, independent of the catalog they may ultimately be a part of.

A CatalogItem, at its core, has a concept identifier, CatalogItemCode, which associates the class to a concept in a controlled terminology or ontology. Apart from a number of metadata attributes, a CatalogItem may optionally be associated with a declarative model definition called a CompositeConstraint that drives the representation of the details of the item in a Computed Physician Order Entry System or as an order entry form tied to this catalog.

Catalogs may contain children catalogs included either by reference or inline depending on the implementation. Subcatalogs allow catalogs to be built in a modular fashion and enable reusability of catalog content shared across catalogs.

The relationship between an Order Service Catalog and the Order Management Service are represented by the relationships between an Order and its associated CatalogEntry as well as between the Requirements class and the CatalogItem class.



Order Catalog

### AbstractEntry

Class in package 'Classes'

An entry in a catalog. Entries may be composite, atomic, or references to other entries.

| ATTRIBUTES |
| --- |
| Identifier : Identifier Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Unique identifier for this instantiation of a catalog entry  [ Is static False. Containment is Not Specified. ] |
| Title : String Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Name assigned to the catalog entry.  [ Is static False. Containment is Not Specified. ] |

### Catalog

Class in package 'Classes'

Catalog of knowledge artifacts, products or services that can be ordered or fulfill an order.

| ATTRIBUTES |
| --- |
| Metadata : CatalogMetadata Public  Supporting information, provenance and attribution metadata for this catalog.  [ Is static False. Containment is Not Specified. ] |

### CatalogEntry

Class in package 'Classes'

A catalog entry is a wrapper for an artifact, product, or service that can be selected, ordered, and exchanged from a catalog. The CatalogEntry class contains metadata that contextualizes a given catalog item within the context of a catalog. Note that attributes that are specific to a catalog item independent of the catalog where this item is referenced belong to the CatalogItem class.

| ATTRIBUTES |
| --- |
| Metadata : CatalogEntryMetadata Public  Information about a catalog object to support the curation and management of the catalog object. Note that catalog entries provide the contextual information associated with an orderable item in the context of the catalog or subcatalog.  [ Is static False. Containment is Not Specified. ] |

| ASSOCIATIONS | |
| --- | --- |
| Association (direction: Unspecified) item  The item referenced by this catalog entry. | |
| Source: Public (Class) CatalogEntry  Cardinality: [0..\*] | Target: Public (Class) CatalogItem  Cardinality: [1] |
| AssociationClass (direction: Source -> Destination) RelatedEntry  Relationship between a catalog entry and another catalog entry. For instance, a catalog entry may be a substitute for another catalog entry in the context of a given catalog. | |
| Source: Public (Class) CatalogEntry  Cardinality: [0..\*] | Target: Public (Class) CatalogEntry  Cardinality: [1] |
| AssociationClass (direction: Source -> Destination) RelatedEntry  Relationship between a catalog entry and another catalog entry. For instance, a catalog entry may be a substitute for another catalog entry in the context of a given catalog. | |
| Source: Public (Class) CatalogEntry  Cardinality: [0..\*] | Target: Public (Class) CatalogEntry  Cardinality: [1] |
| Association (direction: Source -> Destination) isBasedOn | |
| Source: Public (Class) Order  Cardinality: [0..\*] | Target: Public (Class) CatalogEntry  Cardinality: [1] |

### CatalogEntryGroup

Class in package 'Classes'

A collection of catalog entries or nested entry groups that help organize a catalog. Note that catalog entry groups are recursively nested structures. CatalogEntryGroups are equivalent to sections in a document.

### CatalogEntryReference

Class in package 'Classes'

A reference to a catalog entry. This allows a catalog entry to be defined in one part of the catalog and referenced in another part of the catalog.

| ATTRIBUTES |
| --- |
| ReferenceId : VersionedIdentifier Public  A reference to a catalog entry.  [ Is static False. Containment is Not Specified. ] |

### CatalogEntryRelationship

AssociationClass in package 'Classes'

Relationship between a catalog entry and another catalog entry. For instance, a catalog entry may be a substitute for another catalog entry in the context of a given catalog. For example, prerequisite or reflex.

CatalogEntryRelationship

Version 1.0 Phase 1.0 Proposed

Claude Nanjo created on 7/21/2017. Last modified 8/14/2017

| ATTRIBUTES |
| --- |
| relationshipType : CodeableConcept Public  A concept representing the type of relationship between the catalog entries.  [ Is static False. Containment is Not Specified. ] |

### CatalogItem

Class in package 'Classes'

A catalog item represents an artifact, product or a service that can be selected from a catalog. Catalog items are essentially mappings between a catalog item concept (e.g., a code for a laboratory test), the serialization of the object's details, and the model and associated constraints on that model that define its order entry form specification. An example might include a dispensable medication term such as “Amoxicillin 200 mg tablet”; a laboratory test or panel LOINC concept with details about the laboratory test and any order entry attribute definitions and constraints. These may be presented to the user when ordering the item from the catalog. For instance, a medication model associated with "Amoxicillin 200 mg tablet" may specify a 'route of administration' attribute constrained to a subset of possible routes of administration.

| ATTRIBUTES |
| --- |
| Identifier : Identifier Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Unique identifier for this CatalogItem.  [ Is static False. Containment is Not Specified. ] |
| Name : String Public  Human readable name assigned to this catalog item.  [ Is static False. Containment is Not Specified. ] |
| Description : String Public  A text narrative describing the catalog item.  [ Is static False. Containment is Not Specified. ] |
| CatalogItemCode : CodeableConcept Public  Concept representing the item. E.g., "Metoprolol 25 mg tablet".  [ Is static False. Containment is Not Specified. ] |
| Details : WrappedContent Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  A serialized object representing the attributes that comprise up the details of this catalog item. Note that there is a potentially wide variety of catalog item types each with a specific set of attributes.  [ Is static False. Containment is Not Specified. ] |

| ASSOCIATIONS | |
| --- | --- |
| Association (direction: Source -> Destination) orderEntryFormDefinition  Model representing the details of this catalog item for rendering in a CPOE system for instance. CompositeConstraints are typically used to render order detail forms with their associated constraints (e.g., a field for a route of administration and value set constraints to represent the values that can be selected). | |
| Source: Public (Class) CatalogItem  Cardinality: [0..\*] | Target: Public (Class) CompositeConstraint  Cardinality: [0..1] |
| AssociationClass (direction: Source -> Destination) RelatedItem  Catalog items may be related to other catalog items. For instance, a laboratory test is generally associated with a specimen. | |
| Source: Public (Class) CatalogItem  Cardinality: [0..\*] | Target: Public (Class) CatalogItem  Cardinality: [1] |
| Association (direction: Unspecified) item  The item referenced by this catalog entry. | |
| Source: Public (Class) CatalogEntry  Cardinality: [0..\*] | Target: Public (Class) CatalogItem  Cardinality: [1] |
| AssociationClass (direction: Source -> Destination) RelatedItem  Catalog items may be related to other catalog items. For instance, a laboratory test is generally associated with a specimen. | |
| Source: Public (Class) CatalogItem  Cardinality: [0..\*] | Target: Public (Class) CatalogItem  Cardinality: [1] |
| Association (direction: Unspecified) | |
| Source: Public (Class) Requirement  Cardinality: [0..\*] | Target: Public (Class) CatalogItem  Cardinality: [1] |

### CatalogItemRelationship

AssociationClass in package 'Classes'

Catalog items may be related to other catalog items. For instance, a laboratory test is generally associated with a specimen.

| ATTRIBUTES |
| --- |
| RelationshipType : CodeableConcept Public  A concept that represents the relationship. For instance, the relationship between a laboratory test and a specimen may be related with the 'Specimen (attribute)' concept.  [ Is static False. Containment is Not Specified. ] |

### CompositeConstraint

Class in package 'Classes'

A constraint composed of either other composite constraints or atomic constraints. Note that composite constraints are recursive. Generally, composite constraints define a set of constraints that apply to a given catalog item entry form definition. Examples of composite constraints are OpenEHR Archetypes or FHIR StructureDefinitions.

| ASSOCIATIONS | |
| --- | --- |
| Association (direction: Source -> Destination) orderEntryFormDefinition  Model representing the details of this catalog item for rendering in a CPOE system for instance. CompositeConstraints are typically used to render order detail forms with their associated constraints (e.g., a field for a route of administration and value set constraints to represent the values that can be selected). | |
| Source: Public (Class) CatalogItem  Cardinality: [0..\*] | Target: Public (Class) CompositeConstraint  Cardinality: [0..1] |

### Order

Class in package 'Classes'

This class represents the order as a whole. Each order is distinguished by having both a single subject and a single requestor.

| ATTRIBUTES |
| --- |
| OrderIdentities : Identifier Public  Multiplicity: ( [1..\*], Allow duplicates: 0, Is ordered: False )  Unique identifiers for this instantiation of a request for performance of order.  [ Is static False. Containment is Not Specified. ] |
| OrderedBy : ClinicalPractitionerReference Public  Authorized individual responsible for the order.  [ Is static False. Containment is Not Specified. ] |
| OrderTime : DateTime Public  The time the order was released for Fulfillment.  [ Is static False. Containment is Not Specified. ] |
| Status : OrderStatus Public  Overall status of the order.  [ Is static False. Containment is Not Specified. ] |
| InterestedParties : InterestedParty Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Reference(s) to other parties (Clinical Practitioners, Organizations, etc) that have an interest in the outcome of this order.  [ Is static False. Containment is Not Specified. ] |
| OrderProvenance : Provenance Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  The provenance of the order.  [ Is static False. Containment is Not Specified. ] |
| RelatedOrders : OrderRelationship Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Collection of the other orders related to this order.  [ Is static False. Containment is Not Specified. ] |
| ItemGroupPolicy : OrderItemGroupPolicy Public  Policy information on how order item groups interrelate.  [ Is static False. Containment is Not Specified. ] |
| Priority : Coding Public  The criticality of the order.  [ Is static False. Containment is Not Specified. ] |
| Context : OrderContext Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  The context in which this order is occurring.  [ Is static False. Containment is Not Specified. ] |

| ASSOCIATIONS | |
| --- | --- |
| Association (direction: Destination -> Source) | |
| Source: Public OrderIdentity (Class) Order  Cardinality: [0..\*] | Target: Public (Class) OrderResponse |
| Association (direction: Source -> Destination) isBasedOn | |
| Source: Public (Class) Order  Cardinality: [0..\*] | Target: Public (Class) CatalogEntry  Cardinality: [1] |
| Association (direction: Source -> Destination) | |
| Source: Public (Abstract) OrderAction «abstract» | Target: Public Order (Class) Order  Cardinality: [1] |
| Association (direction: Source -> Destination) | |
| Source: Public (Class) ResultAugmentation | Target: Public OrderIdentity (Class) Order  Cardinality: [1] |
| Association (direction: Source -> Destination) | |
| Source: Public (Class) SubstitutionProposal | Target: Public OrderIdentity (Class) Order  Cardinality: [1]  The Order for which this substitution is targeted |
| Association (direction: Destination -> Source) | |
| Source: Public Subject (Class) SubjectReference  Cardinality: [1] | Target: Public (Class) Order  Cardinality: [0..\*] |
| Association (direction: Source -> Destination) | |
| Source: Public (Class) Promise | Target: Public OrderIdentity (Class) Order  Cardinality: [1] |

### Requirement

Class in package 'Classes'

Constraints or preconditions to the acceptance or fulfillment of an order. Requirements can be about the collection of a specimen, counselling, patient presence or procedure / sequence.

| ATTRIBUTES |
| --- |
| Identifier : Identifier Public  The identifier for this requirement. Maybe be used to link to the requirement externally.  [ Is static False. Containment is Not Specified. ] |
| Type : RequirementType Public  Hard Prerequisite - Reject if not present on create.  Additional Workflow (e.g. Analyte collection)  [ Is static False. Containment is Not Specified. ] |
| ContentSignifier : ContentSignifier Public  The nature of the content of this requirement.  [ Is static False. Containment is Not Specified. ] |
| Content : Base64Binary Public  The encoded requirement content.  [ Is static False. Containment is Not Specified. ] |
| Source : RequirementsSource Public  Indication of the source of the requirement. Some example sources include the 0Order Management service might be the source, a workflow and policy engine, specific fulfiller requirements, etc.  [ Is static False. Containment is Not Specified. ] |

| ASSOCIATIONS | |
| --- | --- |
| Association (direction: Unspecified) | |
| Source: Public (Class) Requirement  Cardinality: [0..\*] | Target: Public (Class) CatalogItem  Cardinality: [1] |

### Constraint diagram

Class diagram in package 'Order Catalog'

Orderable items are mappings between an orderable concept and its formal model definition. For instance, medication orderables may be associated with a Medication Orderable Definition which specifies a number of attributes, called Orderable Attribute Definitions. These attributes are used to define the details of a medication orderable and include attribute definitions for attributes such as route, dose quantity, frequency, PRN reason, etc… Orderable definitions can be derived from other orderable definitions and composed of other definitions as well thus supporting more modular designs. For instance, a complex medication pattern may be composed of a number of children patterns.

An Orderable Item constraint is a set of constraints applied on the orderable definition. Constraints are typically used to restrict the range of an attribute such as Value Set Constraints. For instance, a medication orderable may associate with the ‘dose unit’ attribute a large value set of possible unit codes. However, specific medication products may restrict these codes to relevant subsets of the master list. Additional types of constraints include attribute redefinitions conformant with their original specification in the template or in parent constraints such as name changes or the tightening of cardinality constraints and of type constraints.

Constraints may be atomic or composite. This specification defines the set of constraints required for the proper definition of catalog orderable items. Additional constraints may be considered in specific platform specific implementations. For instance, a FHIR-based implementation of the Order Catalog Module may consider the use of Structure Definitions to capture such constraints and template component definitions.



Constraint

### AbstractConstraint

Abstract «abstract» in package 'Classes'

Parent class for model constraints. Constraints may be atomic or composite.

AbstractConstraint

Version 1.0 Phase 1.0 Proposed

Claude Nanjo created on 2/11/2017. Last modified 8/25/2017

| ATTRIBUTES |
| --- |
| Identifier : Identifier Public  Multiplicity: ( [1..\*], Allow duplicates: 0, Is ordered: False )  An identifier for the constraint.  [ Is static False. Containment is Not Specified. ] |
| Name : String Public  The name assigned to this constraint definition  [ Is static False. Containment is Not Specified. ] |

| ASSOCIATIONS | |
| --- | --- |
| Association (direction: Source -> Destination) ReferencedTemplateComponent  The template component this constraint definition constrains. | |
| Source: Public (Abstract) AbstractConstraint «abstract» | Target: Public (Class) TemplateComponent  Cardinality: [1] |
| Association (direction: Source -> Destination) parent  The parent constraint this constraint redefines. Note that constraints can only tighten parent constraint. They may not loosen a parent constraint. | |
| Source: Public (Abstract) AbstractConstraint «abstract» | Target: Public (Abstract) AbstractConstraint «abstract» |
| Association (direction: Source -> Destination) parent  The parent constraint this constraint redefines. Note that constraints can only tighten parent constraint. They may not loosen a parent constraint. | |
| Source: Public (Abstract) AbstractConstraint «abstract» | Target: Public (Abstract) AbstractConstraint «abstract» |

### AtomicConstraint

Abstract «abstract» in package 'Classes'

An atomic constraint constrains a specific aspect of an orderable template. Atomic constraints may be constraints applied on the overall orderable definition or attribute constraints that constrain specific aspects of an orderable detail attribute definition.

### AttributeCardinalityRestriction

Class in package 'Classes'

Constraint refining an attribute's cardinality definition. Note that cardinality constraint must be consistent with the cardinality specification in the attribute definition or in a parent cardinality constraint specification.

| ATTRIBUTES |
| --- |
| min : int Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  The minimum cardinality constraint.  [ Is static False. Containment is Not Specified. ] |
| max : int Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  The maximum cardinality constraint.  [ Is static False. Containment is Not Specified. ] |

### AttributeNameConstraint

Class in package 'Classes'

A constraint applied to the name of an attribute that allows the redefinition of that name to a more context-friendly name.

### AttributeNameRedefine

Class in package 'Classes'

Constraint that redefines the name of an attribute specified in a parent constraint or in the attribute definition.

| ATTRIBUTES |
| --- |
| redefinedName : String Public  The new attribute name value  [ Is static False. Containment is Not Specified. ] |

### AttributeRangeRestriction

Class in package 'Classes'

Constraint that restricts the allowable range of an attribute. For instance, it may constrain a coded attribute to a value set or a decimal type to a range of allowed values on the number line.

### AttributeSemanticBindingConstraint

Class in package 'Classes'

A constraint on an attribute semantic binding.

| ATTRIBUTES |
| --- |
| redefinedSemanticBinding : Coding Public  The concept associated with this attribute. This code must be subsumed by either a parent semantic binding constraint or the semantic binding associated with the attribute at the template component level.  [ Is static False. Containment is Not Specified. ] |

### AttributeTypeRestriction

Class in package 'Classes'

A constraint narrowing the type associated with this attribute definition.

| ATTRIBUTES |
| --- |
| conformsToType : TypeDefinition Public  The type this attribute must conform to. Note that a type constraint must conform to either the parent constraint's type definition or the type definition of the attribute definition.  [ Is static False. Containment is Not Specified. ] |

### AttributeValueConstraint

Class in package 'Classes'

A constraint applied on the value of an attribute definition (as opposed to the predicate name)

### CatalogItemAttributeConstraint

Abstract «abstract» in package 'Classes'

Attribute constraints constrain attribute definitions. The most common types of attribute constraints are cardinality constraints, attribute type constraints, value set constraints, default value constraints, and attribute name redefines.

### CatalogItemAttributeDefinition

Class in package 'Classes'

An attribute definition that must be specified to complete a catalog entry - eg dosage. This defines the template attributes that are instantiated in a specific kind of catalog entry

| ATTRIBUTES |
| --- |
| Value : TypeDefinition Public  The type definition for this attribute.  [ Is static False. Containment is Not Specified. ] |

### CatalogItemConstraint

Abstract «abstract» in package 'Classes'

Template constraint that constrains the overall orderable definition. Typically these consist of invariants, template name redefinitions, or constraints on template semantic bindings.

|  |
| --- |
| Generalization from ClassifierSemanticBindingConstraint to «abstract» CatalogItemConstraint  [ Direction is 'Source -> Destination'. ] |

### CatalogItemDefinition

Class in package 'Classes'

Definitional structure that describes the characteristics that can be defined for a family of catalog items. For instance a medication orderable in a catalog of medication orderables might include dosage, frequency, duration, administration instructions, etc.

| ASSOCIATIONS | |
| --- | --- |
| Association (direction: Source -> Destination) Parent  The parent orderable definition specialized by this definition. | |
| Source: Public (Class) CatalogItemDefinition | Target: Public (Class) CatalogItemDefinition |
| Association (direction: Source -> Destination) Parent  The parent orderable definition specialized by this definition. | |
| Source: Public (Class) CatalogItemDefinition | Target: Public (Class) CatalogItemDefinition |

### ClassifierInvariantConstraint

Class in package 'Classes'

An invariant rule to apply on the orderable definition. This is typically used to specify attribute dependency constraints.

| ATTRIBUTES |
| --- |
| invariant : Invariant Public  An invariant constraint on a classifier.  [ Is static False. Containment is Not Specified. ] |

### ClassifierNameRedefine

Class in package 'Classes'

A class name redefinition constraint. It allows the renaming of an orderable definition to a more context-friendly name.

| ATTRIBUTES |
| --- |
| redefinedName : String Public  Redefinition of the name of a template  [ Is static False. Containment is Not Specified. ] |

### ClassifierSemanticBindingConstraint

Class in package 'Classes'

A constraint on an orderable semantic binding.

| ATTRIBUTES |
| --- |
| redefinedSemanticBinding : Coding Public  A constrained classifier semantic binding. The assigned code must be subsumed by the parent constraint or orderable definition code.  [ Is static False. Containment is Not Specified. ] |

### CodedAttributeRangeConstraint

Class in package 'Classes'

A value set constraint on a coded attribute type.

| ATTRIBUTES |
| --- |
| conformsToValueSet : Identifier Public  The value set identifier defining the range of this coded attribute definition.  [ Is static False. Containment is Not Specified. ] |

### CompositeConstraint

Class in package 'Classes'

A constraint composed of either other composite constraints or atomic constraints. Note that composite constraints are recursive. Generally, composite constraints define a set of constraints that apply to a given catalog item entry form definition. Examples of composite constraints are OpenEHR Archetypes or FHIR StructureDefinitions.

| ASSOCIATIONS | |
| --- | --- |
| Association (direction: Source -> Destination) orderEntryFormDefinition  Model representing the details of this catalog item for rendering in a CPOE system for instance. CompositeConstraints are typically used to render order detail forms with their associated constraints (e.g., a field for a route of administration and value set constraints to represent the values that can be selected). | |
| Source: Public (Class) CatalogItem  Cardinality: [0..\*] | Target: Public (Class) CompositeConstraint  Cardinality: [0..1] |

### ConstrainedTypeConstraint

Class in package 'Classes'

A constraint restriction on an attribute type. For instance, the application of a MyPatientCompositeConstraint on the Patient type.

| ATTRIBUTES |
| --- |
| conformsToConstraint : AbstractConstraint Public  The constraint that this attribute type must conform to.  [ Is static False. Containment is Not Specified. ] |

### TemplateComponent

Class in package 'Classes'

A component of an orderable item model.

| ATTRIBUTES |
| --- |
| Identifier : Identifier Public  Unique identifier for the template component.  [ Is static False. Containment is Not Specified. ] |
| Name : String Public  Textual description of the template component.  [ Is static False. Containment is Not Specified. ] |
| Metadata : CatalogMetadata Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Information about a catalog object to support the curation and management of the catalog object  [ Is static False. Containment is Not Specified. ] |
| SemanticBinding : CodeableConcept Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Concept identifying the specific template component.  [ Is static False. Containment is Not Specified. ] |

| ASSOCIATIONS | |
| --- | --- |
| Association (direction: Source -> Destination) ReferencedTemplateComponent  The template component this constraint definition constrains. | |
| Source: Public (Abstract) AbstractConstraint «abstract» | Target: Public (Class) TemplateComponent  Cardinality: [1] |

### Catalog Support diagram

Class diagram in package 'Order Catalog'

The Catalog Support package consists of supporting data structures to support the definition of the catalog model and its operations.



Catalog Support

### List

Class in package 'Classes'

A collection of information.

| ATTRIBUTES |
| --- |
| Items : T Public  Template for order collections.  [ Is static False. Containment is Not Specified. ] |

### CatalogEntryGroupList

Class in package 'Classes'

A list of Catalog Entry Groups

### CatalogEntryList

Class in package 'Classes'

Class representing a list of catalog entries.

### CatalogEntryMetadata

Class in package 'Classes'

Information about a catalog entry object to support the curation and management of the catalog object.

| ATTRIBUTES |
| --- |
| Identifier : VersionedIdentifier Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  An identifier for the catalog entry.  [ Is static False. Containment is Not Specified. ] |
| Contributor : Contributor Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Person or organization contributing to the catalog entry. Examples are author, reviewer, etc  [ Is static False. Containment is Not Specified. ] |
| ServiceProvider : Reference Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  The provider(s) of the requested service referenced by this catalog entry.  [ Is static False. Containment is Not Specified. ] |
| Version : SemanticVersion Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Current semantic version of the catalog object  [ Is static False. Containment is Not Specified. ] |
| Status : CodeableConcept Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  The status of the item, e.g. active, approved, deleted…  [ Is static False. Containment is Not Specified. ] |
| ItemType : CodeableConcept Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  The type of item - medication, device, service, protocol or other  [ Is static False. Containment is Not Specified. ] |
| Category : CodeableConcept Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Assigned category for the catalog entry used for domain specific categorizations. Eg Microbiology/ general lab for Lab directory of service; or Medication class for drug formularies  [ Is static False. Containment is Not Specified. ] |
| EntryType : CodeableConcept Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Whether the entry represents an orderable item, or supporting definitional item  [ Is static False. Containment is Not Specified. ] |
| ValidityPeriod : Period Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  The time range when this catalog entry is valid.  [ Is static False. Containment is Not Specified. ] |

### CatalogEntryReferenceList

Class in package 'Classes'

A list of Catalog Entry References.

### CatalogItemDefinitionList

Class in package 'Classes'

Class representing a list of catalog item definition templates.

### CatalogItemList

Class in package 'Classes'

Class representing a list of catalog items.

### CatalogList

Class in package 'Classes'

Class representing a list of catalogs.

### CatalogMetadata

Class in package 'Classes'

Information about a catalog object to support the curation and management of the catalog object.

| ATTRIBUTES |
| --- |
| Identifier : VersionedIdentifier Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  An identifier for the catalog instance.  [ Is static False. Containment is Not Specified. ] |
| Contributor : Contributor Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Person or organization contributing to the catalog. Examples are author, reviewer, etc ...  [ Is static False. Containment is Not Specified. ] |
| Version : SemanticVersion Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Current semantic version of the catalog object.  [ Is static False. Containment is Not Specified. ] |
| Status : CodeableConcept Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  The business state of the catalog object. For instance, draft, published, deprecated.  [ Is static False. Containment is Not Specified. ] |
| Publisher : Reference Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Person or organization that is the source of the catalog object.  [ Is static False. Containment is Not Specified. ] |
| Copyright : String Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Declaration of the ownership rights / license for the catalog object.  [ Is static False. Containment is Not Specified. ] |
| Title : String Public  Human readable short description of the catalog object.  [ Is static False. Containment is Not Specified. ] |
| SubTitle : String Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Additional qualifying human readable description information.  [ Is static False. Containment is Not Specified. ] |
| validityPeriod : Period Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  The validity period for the catalog.  [ Is static False. Containment is Not Specified. ] |

### Channel

Class in package 'Classes'

How the result of an operation is to be handled upon job initiation. Three options are possible:

1. Stream to a REST endpoint URI

2. Push using web socket technology

3. Persist on server for later retrieval using the job's handle.

Implementations may support one or more of these options.

| ATTRIBUTES |
| --- |
| Type : ChannelType Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  rest-hook | websocket  [ Is static False. Containment is Not Specified. ] |
| Endpoint : URI Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  The endpoint URI for streaming.  [ Is static False. Containment is Not Specified. ] |
| Header : String Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Headers depending on type of channel.  [ Is static False. Containment is Not Specified. ] |
| OutputFormat : ContentSignifier Public  Identifies the desired serialized format for the extract.  [ Is static False. Containment is Not Specified. ] |
| Compression : boolean Public  Boolean indicating whether the extract should be compressed.  [ Is static False. Containment is Not Specified. ] |

### CompositeConstraintList

Class in package 'Classes'

Class representing a list of catalog constraints.

### Contributor

Class in package 'Classes'

Person or organization contributing to the catalog. Examples are author, reviewer, etc.

| ATTRIBUTES |
| --- |
| Type : ContributorType Public  The role performed by a person or organization with respect to the catalog object  [ Is static False. Containment is Not Specified. ] |
| ContributorReference : Reference Public  A reference to the contributor. Includes identifier information as well as content based identification for error checking.  [ Is static False. Containment is Not Specified. ] |

### ExportOptions

Class in package 'Classes'

Configuration parameters for catalog exports.

| ATTRIBUTES |
| --- |
| CatalogIdentifier : VersionedIdentifier Public  The version of the catalog to export.  [ Is static False. Containment is Not Specified. ] |
| IsPartial : boolean Public  Boolean indicating whether a full or partial extract is required. When a full extract is specified, query and catalog category attributes will be ignored.  [ Is static False. Containment is Not Specified. ] |
| Query : Query Public  Multiplicity: ( [0..\*], Allow duplicates: 0, Is ordered: False )  Query to be used when wishing to select only a subset of catalog items. Note that this attribute is ignored if isPartial is set to false.  [ Is static False. Containment is Not Specified. ] |
| NotificationURL : URI Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  An optional URL of a notification callback (CatalogTransferCallback)  [ Is static False. Containment is Not Specified. ] |
| OutputChannel : Channel Public  Channel for the export data stream.  [ Is static False. Containment is Not Specified. ] |

### ExportResult

Class in package 'Classes'

Return value for a catalog export operation including the extract and an operation result representing success or failure of the operation.

| ATTRIBUTES |
| --- |
| Extract : WrappedContent Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Serialized catalog content resulting from an export operation  [ Is static False. Containment is Not Specified. ] |
| OperationResult : OperationResult Public  General operation returned structure to indicate success or failure and provide detailed information in the event of an operations failure.  [ Is static False. Containment is Not Specified. ] |
| JobId : String Public  The identifier for the batch job that triggered this extract.  [ Is static False. Containment is Not Specified. ] |

### ImportOptions

Class in package 'Classes'

Configuration parameters for catalog imports.

| ATTRIBUTES |
| --- |
| FailOnError : boolean Public  Boolean indicating whether the import should abort or continue when an error is encountered.  [ Is static False. Containment is Not Specified. ] |
| InputChannel : Channel Public  Channel for the import data stream.  [ Is static False. Containment is Not Specified. ] |
| NotificationURL : URI Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  An optional URL of a notification callback (CatalogTransferCallback)  [ Is static False. Containment is Not Specified. ] |

### ImportResult

Class in package 'Classes'

Return value for a catalog import operation including the report and an operation result representing success or failure of the operation.

| ATTRIBUTES |
| --- |
| ImportReport : WrappedContent Public  Multiplicity: ( [0..1], Allow duplicates: 0, Is ordered: False )  Report summarizing the results of a catalog import operation  [ Is static False. Containment is Not Specified. ] |
| OperationResult : OperationResult Public  General operation returned structure to indicate success or failure and provide detailed information in the event of an operations failure.  [ Is static False. Containment is Not Specified. ] |
| JobId : String Public  The identifier for the job that triggered this extract.  [ Is static False. Containment is Not Specified. ] |

### Invariant

Class in package 'Classes'

A constraint rule that applies to a structure.

17

| ELEMENTS OWNED BY Invariant |
| --- |
| JobHandleList : Class |

| ATTRIBUTES |
| --- |
| language : Coding Public  The invariant rule language  [ Is static False. Containment is Not Specified. ] |
| rule : String Public  The invariant rule applied to this classifier.  [ Is static False. Containment is Not Specified. ] |

#### JobHandleList

Class owned by 'Invariant', in package 'Classes'

A list of job handles.

### JobHandle

Class in package 'Classes'

A handle for a batch job performed asynchronously. The handle allows a job to be retrieved once processing is complete.

| ATTRIBUTES |
| --- |
| JobId : String Public  An identifier for an asynchronous batch job that can be used to retrieve the job.  [ Is static False. Containment is Not Specified. ] |
| JobOwner : Identifier Public  The user who triggered the job execution.  [ Is static False. Containment is Not Specified. ] |
| HandleValidityPeriod : Period Public  The validity period for this handle. This is relevant when the result of the job execution is persisted on the server for later retrieval.  [ Is static False. Containment is Not Specified. ] |
| JobStatus : JobStatusEnumeration Public  The status of the job.  [ Is static False. Containment is Not Specified. ] |
| Channel : Channel Public  The channel assigned to this job.  [ Is static False. Containment is Not Specified. ] |

### Query

Class in package 'Classes'

A query string that adheres to a specific querying language.

| ATTRIBUTES |
| --- |
| QueryLanguage : String Public  The query syntax associated with the query string.  [ Is static False. Containment is Not Specified. ] |
| QueryString : String Public  The query statement  [ Is static False. Containment is Not Specified. ] |

### CatalogResourceType

Enumeration in package 'Classes'

An enumeration of catalog object types that are exposed through the search interface.

| ATTRIBUTES |
| --- |
| CatalogEntry : Public  Enumeration type representing the entries that make up a catalog.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| CatalogEntryReference : Public  Reference to a Catalog Entry  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| CatalogEntryGroup : Public  A Catalog Entry Group.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| CatalogItem : Public  Enumeration type that represents the entity that can be ordered from a catalog. For instance, these include (but are not limited to) order sets, lab tests, and medication products.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| CatalogItemDefinition : Public  Enumeration type representing the structural definition of a catalog item's entry form.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| CompositeConstraint : Public  Enumeration type representing a constraint or set of constraints applied to a template component. Typical examples may be constraints applied on a medication detail form where all coded fields are scoped to the appropriate value sets.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Catalog : Public  An enumeration type representing a catalog of orderable items and supporting entries.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |

### ChannelType

Enumeration in package 'Classes'

The type of export channel.

| ATTRIBUTES |
| --- |
| RestHook : Public  Content streamed to a rest endpoint.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Websocket : Public  Content pushed via a web socket.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Persisted : Public  Content persisted server side for later retrieval upon completion of the job.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |

### ContainerTypeEnum

Enumeration in package 'Classes'

Collection semantic types.

| ATTRIBUTES |
| --- |
| List : Public  Ordered collection with duplicate elements allowed.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Set : Public  Unordered collection that does not allow duplicate elements in the collection.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Bag : Public  Unordered collection allowing duplicate items.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |

### ContributorType

Enumeration in package 'Classes'

Enumerates the roles performed by people and organizations with respect to catalog objects.

| ATTRIBUTES |
| --- |
| Author : Public  The creator of the catalog object  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Editor : Public  Reviews for compliance with catalog guidelines, completeness, etc  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Reviewer : Public  Reviews for clinical appropriateness  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Endorser : Public  The approver of the catalog object  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |

### JobStatusEnumeration

Enumeration in package 'Classes'

Job statuses.

| ATTRIBUTES |
| --- |
| NotStarted : Public  The job was requested and assigned an identifier but has not started.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| InProgress : Public  The job has started but has not yet completed.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Completed : Public  The job has completed successfully.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Paused : Public  The job has been paused.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Cancelled : Public  The job has been cancelled.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |
| Failed : Public  The job has failed and has not completed.  [ Stereotype is «enum». Is static False. Containment is Not Specified. ] |

## Order Catalog Service Definitions

Package in package 'org.omg.orderservice.catalog'

The Order Catalog Management interfaces define a non-exhaustive set of interactions with a product or service catalog. They are grouped as follows:

1. The Order Service Catalog Management interface consists of those operations that support the maintenance of a catalog. These operations typically consist of granular CRUD operations and batch operations such as import and export. This interface supports both read and write access to the catalog.
2. The Order Service Catalog Reporting Interface defines operations for three of the most common reporting requirements. Additional reporting operations may be added in the future.
3. The Order Service Catalog Query interface exposes a subset of the operations present in the Order Service Catalog Management interface that allow the retrieval of catalog items. This interface allows read-only access to a catalog.

Please note that at this time, a catalog eventing interface is considered out of scope but may be considered in future revisions of this specification.

### Catalog Management diagram

Class diagram in package 'Order Catalog Service Definitions'

The Order Service Catalog Management interface enables organizations to interact with and modify their catalog. This interface is also important in allowing organizations to import, export, and synchronize catalog content. For instance, order set content vendors may allow their clients to upload their order sets, orderable catalogs, and orderable models, update that content using the tooling provided by the vendor, and then re-import the modified evidence-based content back into their local catalog.



Catalog Management

### Catalog Query diagram

Class diagram in package 'Order Catalog Service Definitions'



Catalog Query

### Catalog Reporting diagram

Class diagram in package 'Order Catalog Service Definitions'



Catalog Reporting

### Catalog Transfer diagram

Class diagram in package 'Order Catalog Service Definitions'

Interfaces used for transferring catalog entries



Catalog Transfer

### CatalogEntryReferenceReporting

Class in package 'Order Catalog Service Definitions'

Interface for reports pertaining to the CatalogEntryReference resources such as getting the count of CatalogEntryReference in a repository that match the specified search criteria.

### CatalogItemDefinitionReporting

Class in package 'Order Catalog Service Definitions'

Interface for reports pertaining to the CatalogItemDefinition resources such as getting the count of CatalogItemDefinition in a repository that match the specified search criteria.

### CatalogReporting

Class in package 'Order Catalog Service Definitions'

Interface for reports pertaining to the CatalogReporting resources such as getting the count of CatalogReporting in a repository that match the specified search criteria.

### CompositeConstraintReporting

Class in package 'Order Catalog Service Definitions'

Interface for reports pertaining to the CompositeConstraint resources such as getting the count of CompositeConstraint in a repository that match the specified search criteria.

### CatalogEntryGroupManagement

Interface in package 'Order Catalog Service Definitions'

Binding of the Order Service Catalog Management interface to handle Catalog Entry Groups.

### CatalogEntryGroupQuery

Interface in package 'Order Catalog Service Definitions'

This interface is a binding of the Order Service Catalog. Query interface to handle Catalog Entry Group.

### CatalogEntryGroupReporting

Interface in package 'Order Catalog Service Definitions'

Interface for reports pertaining to the CatalogEntryGroup resources such as getting the count of CatalogEntryGroup in a repository that match the specified search criteria.

### CatalogEntryListCatalogManagement

Interface in package 'Order Catalog Service Definitions'

Binding of the Order Service Catalog Management interface to handle CatalogEntryLists.

### CatalogEntryManagement

Interface in package 'Order Catalog Service Definitions'

Binding of the Order Service Catalog Management interface to handle Catalog Entries.

### CatalogEntryQuery

Interface in package 'Order Catalog Service Definitions'

This interface is a binding of the Order Service Catalog. Query interface to handle Catalog Entries.

### CatalogEntryReferenceManagement

Interface in package 'Order Catalog Service Definitions'

Binding of the Order Service Catalog Management interface to handle Catalog Entry References.

### CatalogEntryReferenceQuery

Interface in package 'Order Catalog Service Definitions'

This interface is a binding of the Order Service Catalog. Query interface to handle Catalog Entry Reference.

### CatalogEntryReporting

Interface in package 'Order Catalog Service Definitions'

Interface for reports pertaining to the CatalogEntry resources such as getting the count of CatalogEntry in a repository that match the specified search criteria.

### CatalogItemDefinitionManagement

Interface in package 'Order Catalog Service Definitions'

Binding of the Order Service Catalog Management interface to handle Catalog Item Defintions.

### CatalogItemDefinitionQuery

Interface in package 'Order Catalog Service Definitions'

This interface is a binding of the Order Service Catalog. Query interface to handle catalog item entry form models.

### CatalogItemManagement

Interface in package 'Order Catalog Service Definitions'

Binding of the Order Service Catalog Management interface to handle Orderable Items.

### CatalogItemQuery

Interface in package 'Order Catalog Service Definitions'

This interface is a binding of the Order Service Catalog. Query interface to handle Catalog Items.

### CatalogItemReporting

Interface in package 'Order Catalog Service Definitions'

Interface for reports pertaining to the CatalogItem resources such as getting the count of CatalogItem in a repository that match the specified search criteria.

### CatalogManagement

Interface in package 'Order Catalog Service Definitions'

Binding of the Order Service Catalog Management interface to handle Catalogs.

### CatalogQuery

Interface in package 'Order Catalog Service Definitions'

A query for one or more catalogs.

### CatalogTransfer

Interface in package 'Order Catalog Service Definitions'

The Catalog Transfer interface defines the set of operations used import and export selected sections of the catalog.

| OPERATIONS |
| --- |
| Export (Source : URI , ExportOptions : ExportOptions ) : JobHandle Public  This operation shall initiate an asynchronous operation to query the Order Catalog (OC) for the resources to export based on the export options specified. Depending on the channel selected in the export options and/or supported by the implementation, the export results may be streamed, pushed, or pulled.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| Import (Target : URI , ImportOptions : ImportOptions , CatalogExtract : CatalogExtract ) : JobHandle Public  Method triggers the asynchronous import into an order catalog of the resources specified in the extract. Import may also update catalog metadata based on the extract. Depending on the channel selected in the export options and/or supported by the implementation, the import results may be streamed, pushed, or pulled.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| GetExportResult (exportHandle : JobHandle ) : ExportResult Public  Method returns the ExportResult associated with the given export job handle. If the job is in progress the extract will be null and an operation result should indicate that the job is still in progress. Must be implemented if a ChannelType of PERSISTED is supported by the implementation.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| GetImportResult (ImportHandle : JobHandle ) : ImportResult Public  Method returns the ImportResult associated with the given import job handle. If the import is still in progress, implementations may return null for the import report or the report up to the point the query was made. Must be implemented if a ChannelType of PERSISTED is supported by the implementation.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| FindJobsForUser (User : Identifier , JobStatus : JobStatusEnumeration ) : JobHandleList Public  Method returns all jobs with the given status associated with the user.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |

### CatalogTransferCallback

Interface in package 'Order Catalog Service Definitions'

A Call back interface that a client to the order catalog service may implement to receive transfer status information on.

| OPERATIONS |
| --- |
| OnExportCompletion (exportJobHandle : JobHandle , channel : Channel ) : void Public  Callback method invoked upon the completion of a catalog export. The argument to the method is the job handle for the asynchronous export job of interest and the channel where the job handle is echoed upon completion of export. Only implemented for servers that support a ChannelType of PERSISTED.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| OnImportCompletion (ImportJobHandle : JobHandle , channel : Channel ) : void Public  Callback method invoked upon the completing of a catalog import. The argument to the method is the job handle for the asynchronous export job of interest and the channel where the job handle is echoed upon completion of export. Only implemented for servers that support a ChannelType of PERSISTED.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |

### CompositeConstraintManagement

Interface in package 'Order Catalog Service Definitions'

Binding of the Order Service Catalog Management interface to handle composite constraints.

### CompositeConstraintQuery

Interface in package 'Order Catalog Service Definitions'

A query for a specific composite constraint defined in a catalog.

### OrderServiceCatalogManagement

Interface in package 'Order Catalog Service Definitions'

The Order Service catalog interface is oriented toward consumers which build and maintain the catalog. This is a template interface for entities that require catalog management.

| OPERATIONS |
| --- |
| FindByIdentifier (Identifier : Identifier ) : T Public  This operation shall query the Order Catalog (OC) for the entity of Type T with the matching Identifier. The most recent version is returned.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| FindByIdentifierAndVersion (Identifier : VersionedIdentifier ) : T Public  This operation shall query the Order Catalog (OC) for the entity of Type T with the matching Identifier and version.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| FindInCatalogByIdentifierAndVersion (Catalog : VersionedIdentifier , Identifier : VersionedIdentifier ) : T Public  This operation shall query the Order Catalog (OC) for the entity of Type T with the matching Identifier and version in the context of a specific catalog.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| FindByIdentifierWithHistory (Identifier : Identifier ) : L Public  This operation shall query the Order Catalog (OC) for the entity of Type T with the matching Identifier and return a list of all versions.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| Create (Resource : T ) : VersionedOperationResult Public  The operation creates a new resource of type T in the order catalog.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| Update (Identifier : Identifier , Resource : T ) : VersionedOperationResult Public  This operation updates a catalog resource in the Order Catalog (OC).  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| Delete (Identifier : VersionedIdentifier ) : OperationResult Public  This operation shall query the Order Catalog (OC) to identify the resource to delete. The resource is then deleted from the catalog.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| Search (Parameters : SearchParameterList , omitDetails : boolean ) : L Public  This operation shall query the Order Catalog (OC) for the subset of resources of type T that match one or more of the parameter arguments to the operation.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| Search (Catalog : VersionedIdentifier , Parameters : SearchParameterList , omitDetails : boolean ) : L Public  This operation shall query the Order Catalog (OC) for the subset of resources of type T that match one or more of the parameter arguments to the operation. The search is constrained to the resources contained within a specific version of a catalog.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |

### OrderServiceCatalogQuery

Interface in package 'Order Catalog Service Definitions'

The Order Service Catalog Query Service is oriented towards consumers which are consuming catalog content for creating orders or for systems that wish to interrogate the components that make up a catalog. This is a template interface.

Note that catalog vocabulary queries should rely on the OMG CTS 2 specification or related terminology service definitions.

| OPERATIONS |
| --- |
| FindByIdentifier (Identifier : Identifier ) : T Public  This operation shall query the Order Catalog (OC) for the entity of Type T with the matching Identifier. The most recent version is returned.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |
| Search (Parameters : SearchParameterList ) : L Public  This operation shall query the Order Catalog (OC) for the subset of resources of type T that match one or more of the parameter arguments to the operation.  For catalog component searches restricted to a specific catalog, a search parameter limiting the search to that specific catalog should be provided.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |

### OrderServiceCatalogReporting

Interface in package 'Order Catalog Service Definitions'

| OPERATIONS |
| --- |
| GetItemCount (Parameters : SearchParameterList ) : int Public  This operation shall query the Order Catalog (OC) for a given entity given the search parameters and return the cardinality of the result set.  [ Is static False. Is abstract False. Is return array False. Is query False. Is synchronized False. ] |