

Documentation of Data Representations

A proposed scheme for documenting data structures and vocabularies for machine applications.

Version 1.0 • Proposed



Date/Time Generated:

1/20/2016 1:19:07 PM

Author:

Stephen M Richard

EA Repository : C:\Workspace\Projects\RDA ResearchDataAlliance\DataModel12.eap

CREATED WITH  **ENTERPRISE
ARCHITECT**

Table of Contents

| | |
|--|-----------|
| Table of Contents | 2 |
| 1 Introduction | 4 |
| 1.1 Target applications:..... | 4 |
| 2 Draft Data Type model | 4 |
| 2.1 Diagrams | 4 |
| 2.1.1 Conceptual representation diagram..... | 4 |
| 2.1.2 Concepts diagram | 5 |
| 2.1.3 Data Element Concept diagram | 5 |
| 2.1.4 DataObject Overview diagram | 6 |
| 2.1.5 Data Object diagram | 7 |
| 2.1.6 Data Element diagram | 7 |
| 2.1.7 Value Domain diagram..... | 8 |
| 2.1.8 ArrayVariable diagram | 8 |
| 2.1.9 Physical Implementation diagram..... | 9 |
| 2.2 Classes | 9 |
| 2.2.1 ArrayDimension | 9 |
| 2.2.2 ArrayVariable | 9 |
| 2.2.3 Concept | 10 |
| 2.2.4 ConceptScheme | 10 |
| 2.2.5 ConceptualDomain | 11 |
| 2.2.6 Constraint..... | 11 |
| 2.2.7 ControlledVocabulary..... | 12 |
| 2.2.8 DataElement | 12 |
| 2.2.9 DataMetaAttributes..... | 13 |
| 2.2.10 DataObject | 13 |
| 2.2.11 EnumeratedValueDomain..... | 14 |
| 2.2.12 Example | 14 |
| 2.2.13 ImplementationElement..... | 14 |
| 2.2.14 ImplementationObject | 15 |
| 2.2.15 InterchangeFormat..... | 15 |
| 2.2.16 LogicalDataType | 16 |
| 2.2.17 MeasureClass..... | 16 |
| 2.2.18 MetaAttribute..... | 16 |
| 2.2.19 ObjectClass | 17 |
| 2.2.20 ObjectDataType..... | 17 |
| 2.2.21 ProcessingStep..... | 17 |
| 2.2.22 Property | 17 |
| 2.2.23 RangeConstraint | 18 |
| 2.2.24 SyntacticDataType..... | 18 |
| 2.2.25 Term..... | 19 |
| 2.2.26 UnitOfMeasure | 19 |
| 2.2.27 ValueDomain..... | 19 |
| 3 HelperClasses..... | 21 |
| 3.1 Diagrams | 21 |

| | | |
|----------|--------------------------------|-----------|
| 3.1.1 | Agent-Event-Link diagram | 21 |
| 3.1.2 | Citation diagram | 22 |
| 3.2 | Classes | 22 |
| 3.2.1 | Address | 22 |
| 3.2.2 | Agent | 22 |
| 3.2.3 | Citation | 23 |
| 3.2.4 | Event..... | 23 |
| 3.2.5 | Link..... | 23 |
| 3.2.6 | LinkedAPIDoc | 24 |
| 3.2.7 | Parameter | 24 |
| 3.2.8 | QualifiedAttribution | 25 |
| 3.2.9 | ScopedIdentifier..... | 25 |
| 4 | RDA DataType model..... | 26 |
| 4.1 | DataTypeModel diagram | 26 |
| 4.2 | Contributor..... | 26 |
| 4.3 | DataType..... | 26 |
| 4.4 | Property..... | 27 |
| 4.5 | Provenance..... | 27 |
| 4.6 | Relationship | 27 |
| 4.7 | Representation | 28 |
| 4.8 | Standard | 28 |
| 4.9 | Usage | 28 |

1 Introduction

The scope of this model is the formal representation of information objects that are the basic units of data representation in computer information systems. The model specifies the concept of a DataObject ('type', 'entity', 'object', etc.) that has a collection of attributes, with domains and cardinalities for those attributes, constituting the representation of instances of that type/entity. The model distinguishes the conceptual level definition of these objects and attributes from the implementation of those concepts with a particular representation. Description and documentation of the conceptual level (ObjectClass and Properties) is important for interfaces through which domain practitioners interact with data. Description and documentation of the implementation level (DataObject and DataElements) is important for software systems that automate operations on the data. Representation of the conceptual objects might be implemented as JSON objects, XML elements, rows in a relation, RDF graphs etc.

This model is a synthesis of a variety of existing models for documenting schema and vocabulary used to define representations of information about entities of interest in the world. Inputs include ISO19110, ISO19115, ISO11179, OGC10-090r3 (NetCDF common data model) and the RDA data Type registry prototype (WG output, March 2015).

1.1 Target applications:

- Reference for communities to document the meaning of entities and attributes in data that they share.
- Discover existing data type and attribute definitions for use in constructing data models, to foster interoperability.
- Machine-assisted data integration, based on identification of matching or 'integratable' attribute content.
- Validation of data instances against a type definition.
- Tools that spin up a UI for a particular data type.

2 Draft Data Type model

This section presents a proposed model for representing schema for structured data. Figure 1 presents all the aspects of the model in one summary figure intended to serve as a quick reference to the entire model. The following sections present views focused on particular elements to facilitate understanding the model. It is recommended that one study the detail diagrams first and then return to this summary diagram after studying the different simplified views. The following section **Error! Reference source not found. Error! Reference source not found.** describes each class in the model, listed in alphabetic order.

2.1 Diagrams

2.1.1 Conceptual representation diagram

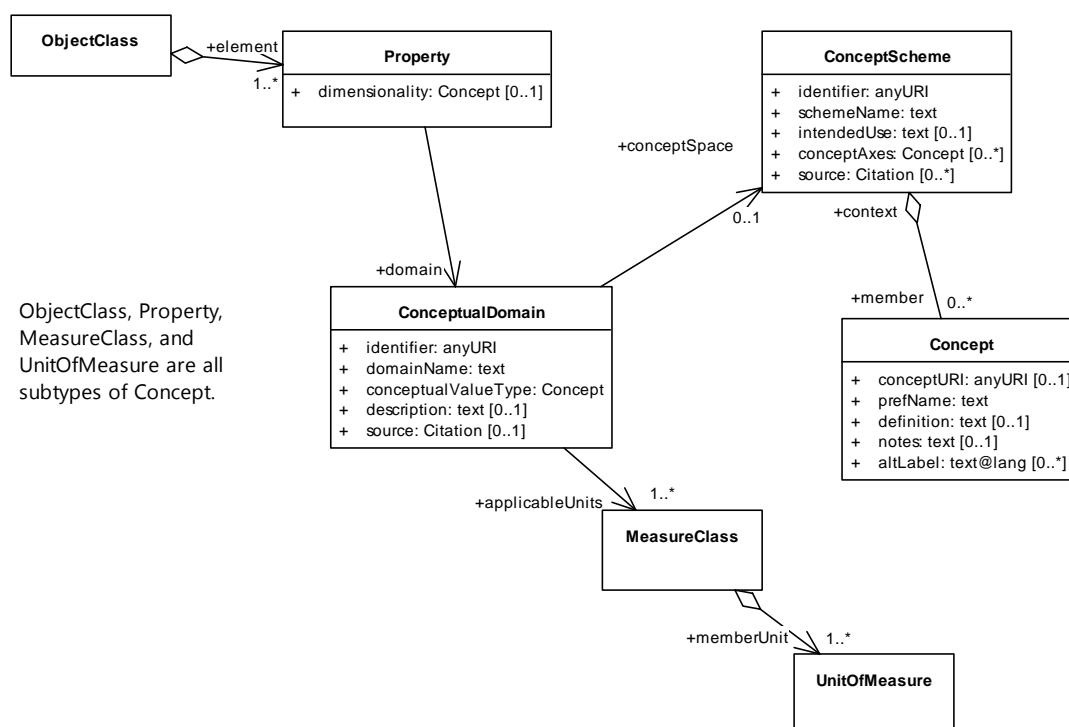


Figure 1: Conceptual representation

2.1.2 Concepts diagram

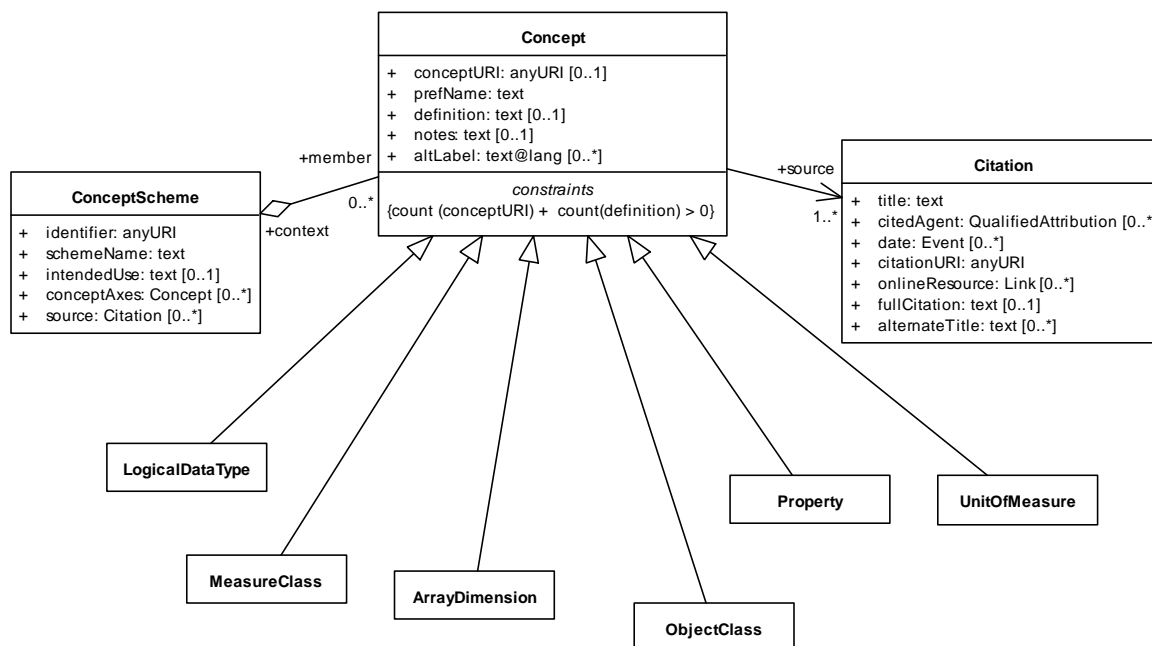


Figure 2: Concepts

2.1.3 Data Element Concept diagram

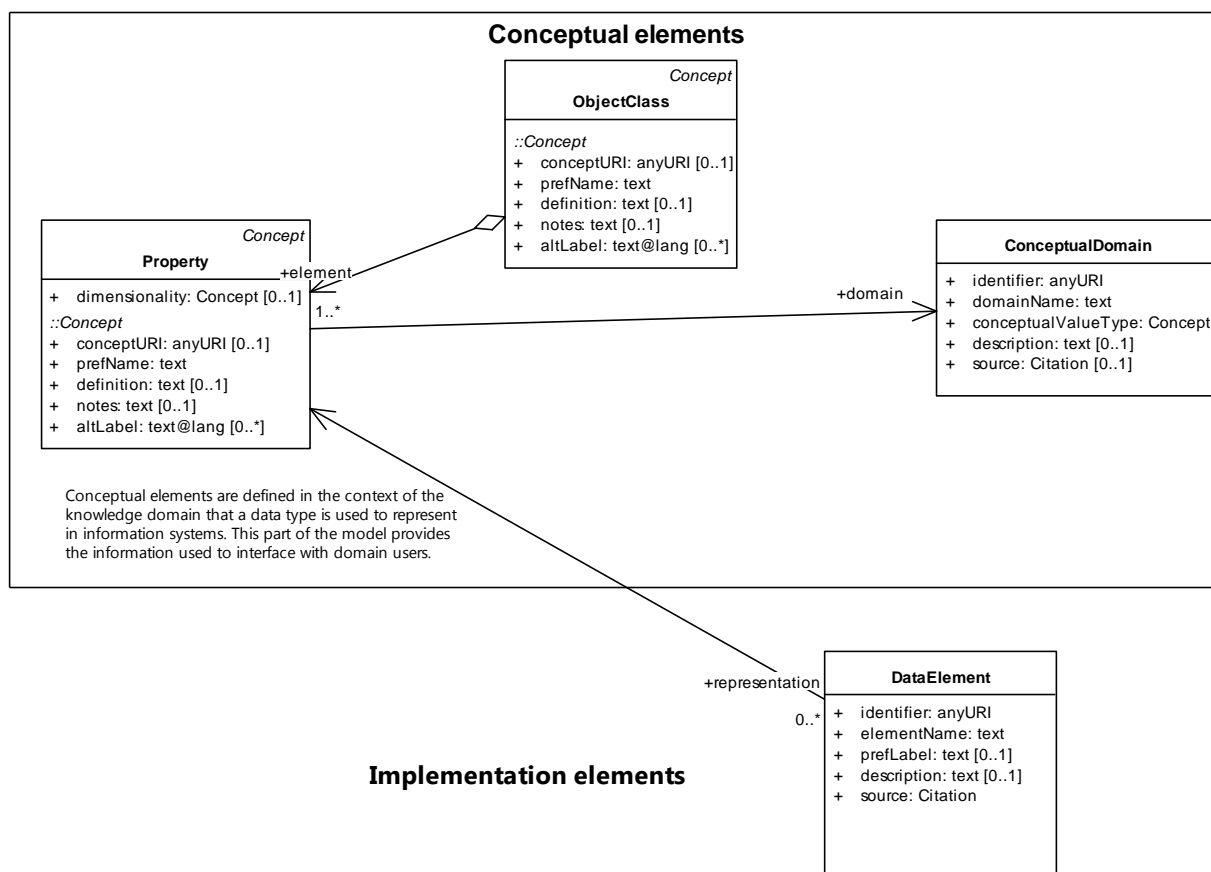


Figure 3: Context:DataElementConcept

2.1.4 DataObject Overview diagram

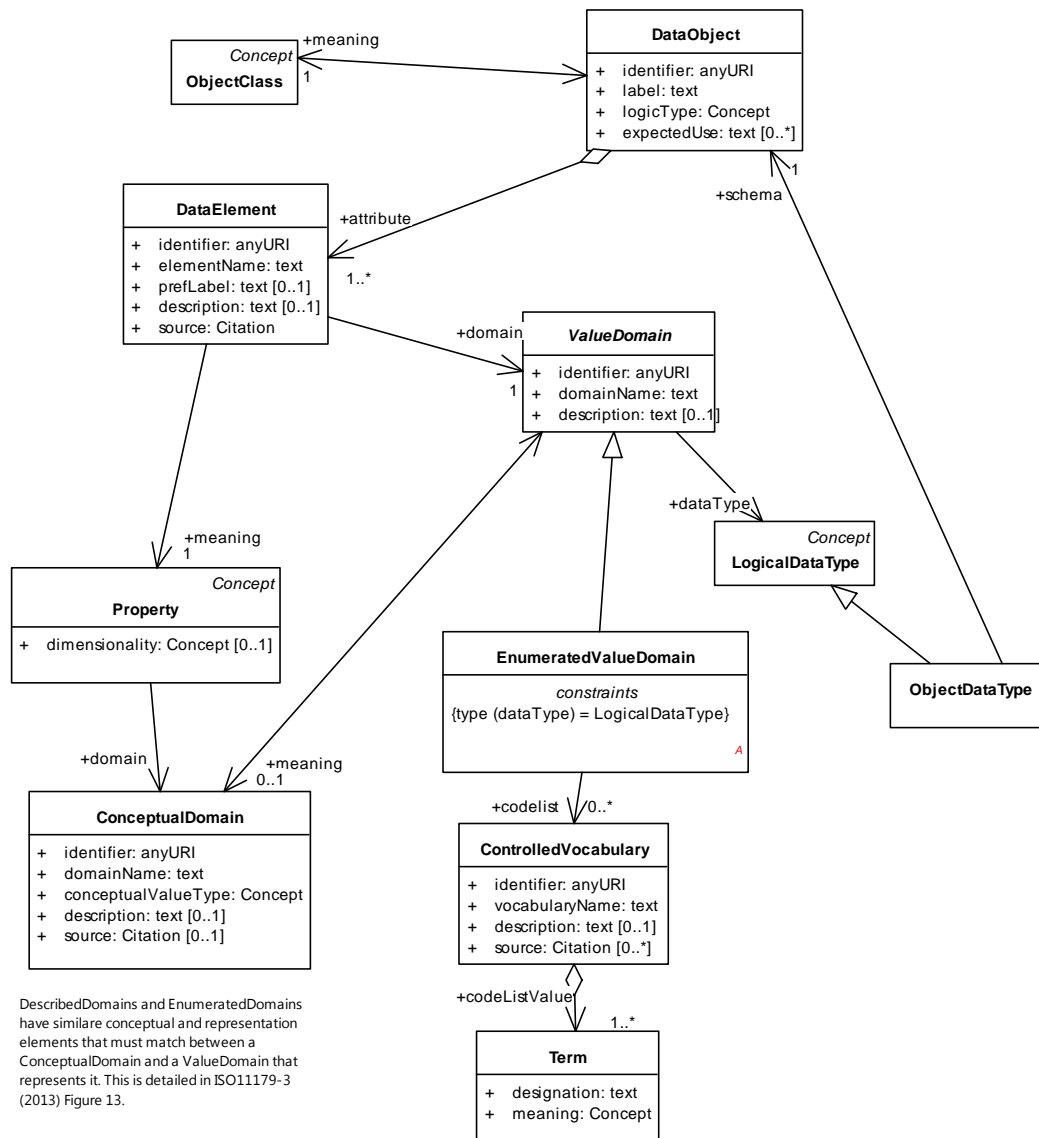


Figure 4: DataObjectOverview

2.1.5 Data Object diagram

A **DataObject** is a **DataType** that provides an implementable representation of an **ObjectClass**. The **ObjectClass** represents the concept of some entity in a domain of interest that is to be represented in an information system.

The attribute **DataElements** associated with the **DataObject** SHALL implement **DataElementConcepts** that fill element roles from the **ObjectClass** represented by the **DataObject**.

Example and **ProcessingStep** are optional content that provides details about the object. These are defined by the **RDA DataType** model.

Conceptual

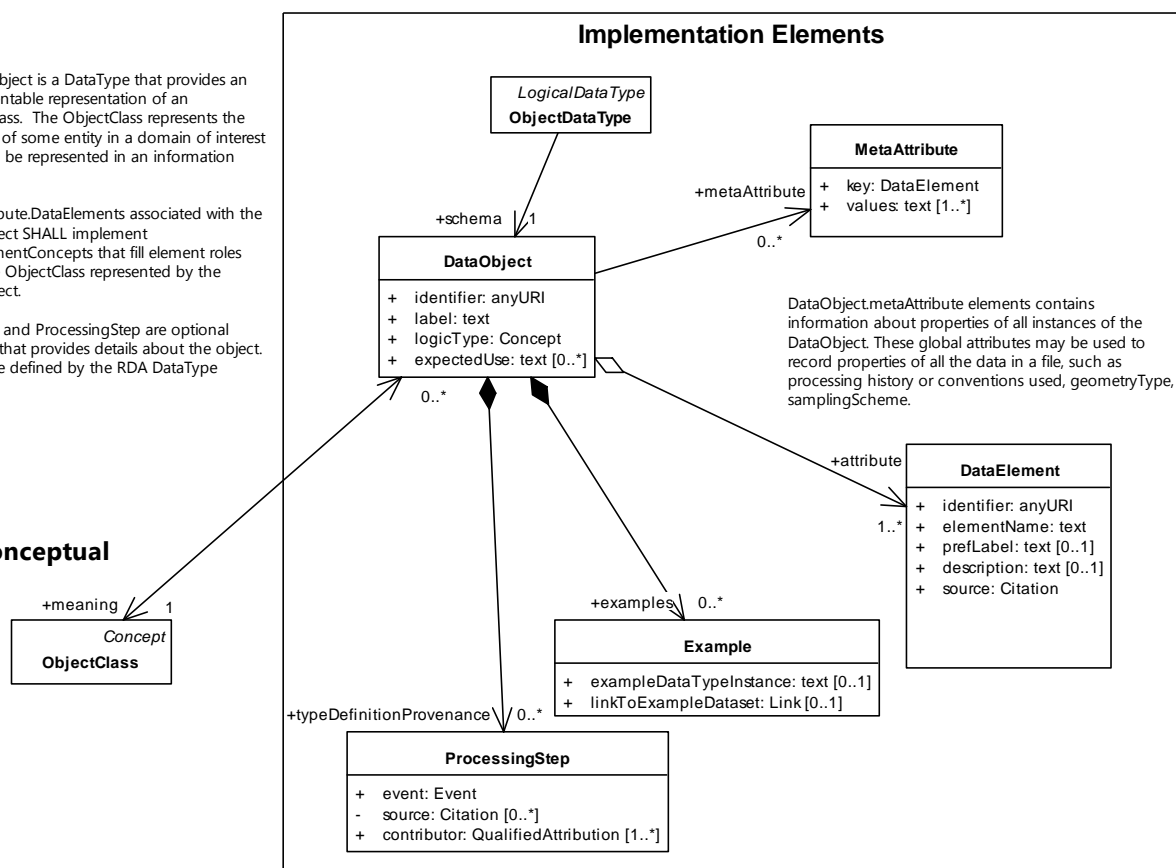


Figure 5: Context:DataObject

2.1.6 Data Element diagram

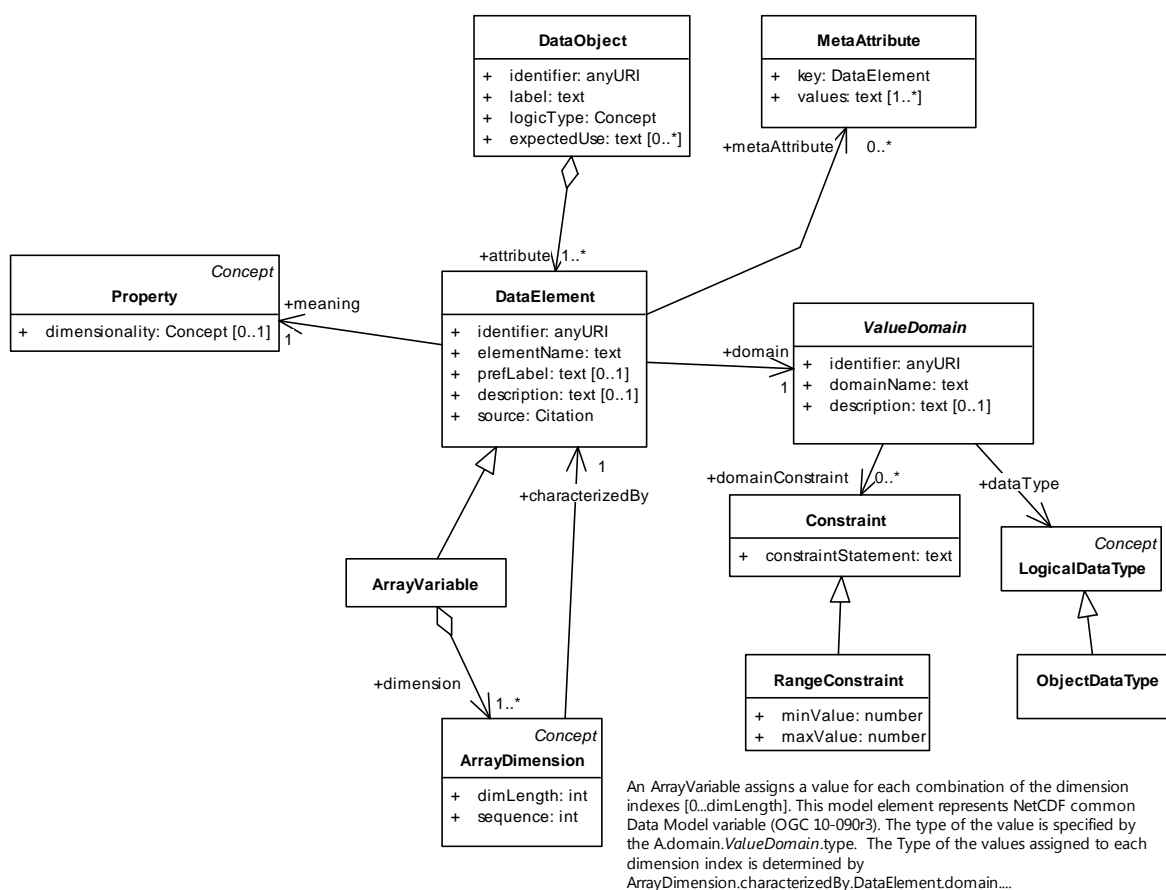


Figure 6: Context:DataElement

2.1.7 Value Domain diagram

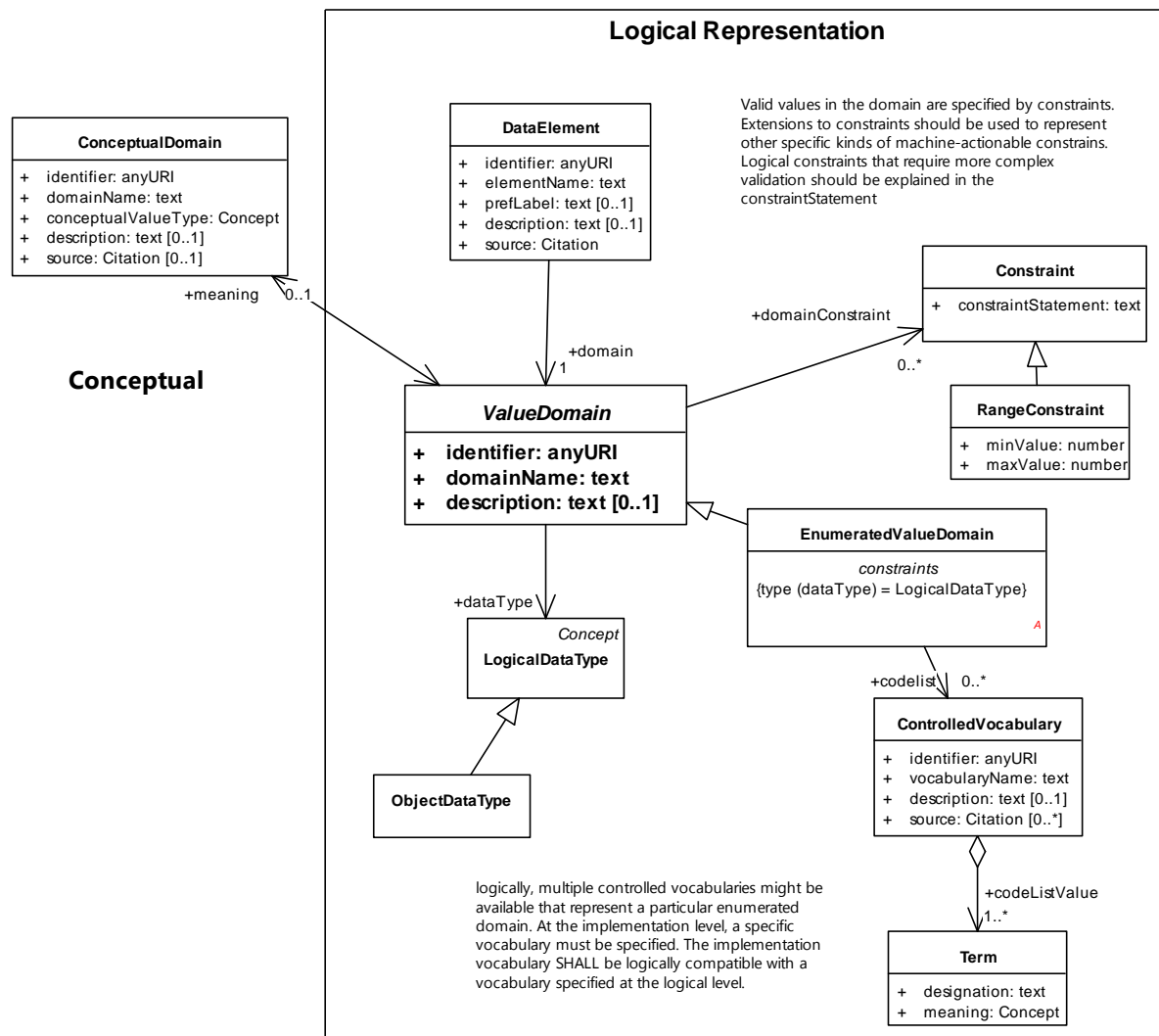


Figure 7: Context:ValueDomain

2.1.8 ArrayVariable diagram

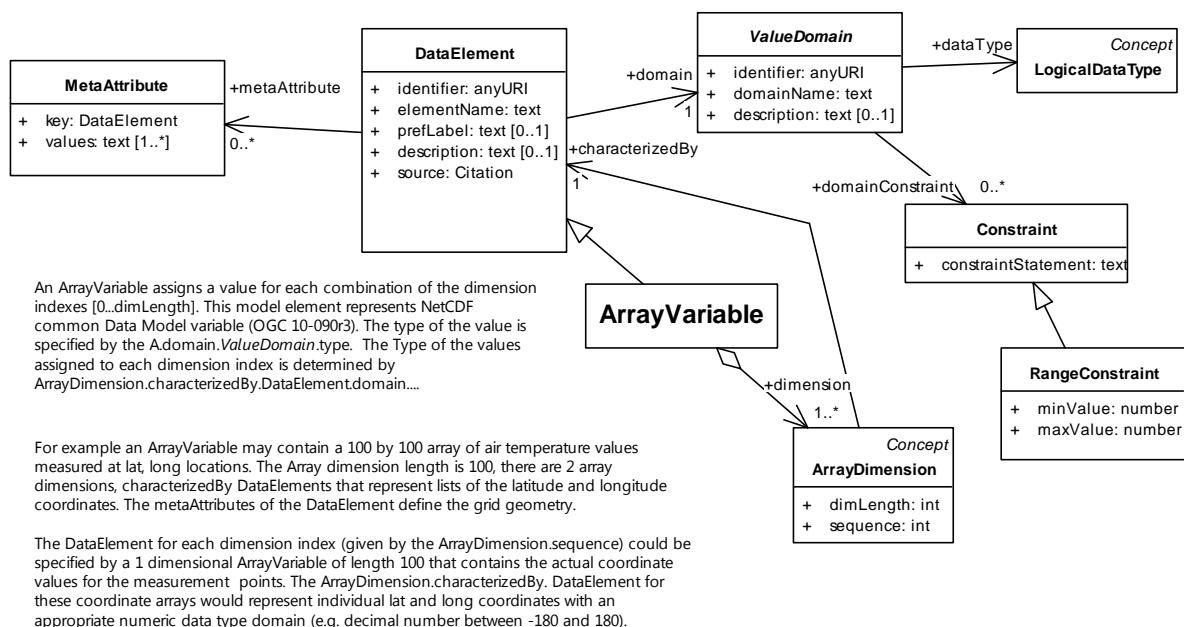


Figure 8: Context:ArrayVariable

2.1.9 Physical Implementation diagram

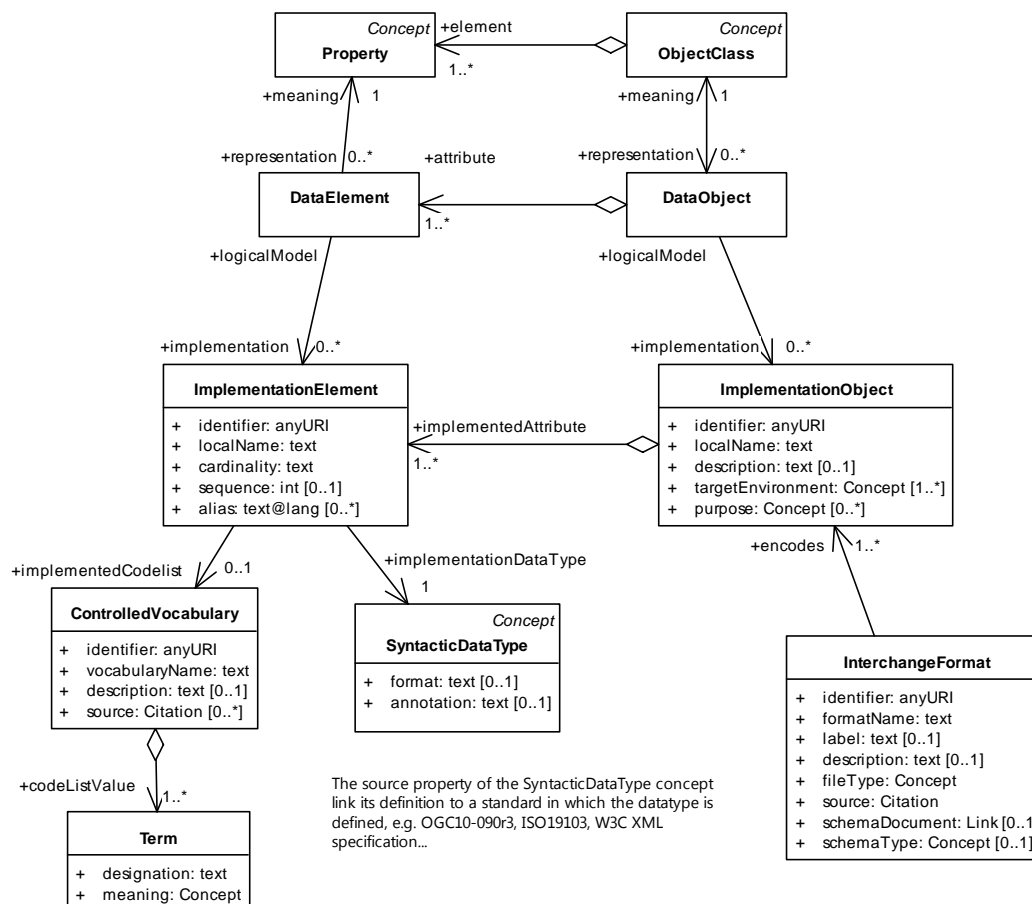


Figure 9: PhysicalImplementation

2.2 Classes

2.2.1 ArrayDimension

NetCDF common data model 'Dimension'. One of the dimensions of an array, has an associated dataTypeValue that assigns meaning.

| OUTGOING STRUCTURAL RELATIONSHIPS | |
|---|---|
| Generalization from ArrayDimension to Concept[Direction is 'Source -> Destination'.] | |
| Aggregation from ArrayDimension to ArrayVariable[Direction is 'Destination -> Source'.] | |
| ATTRIBUTES | |
| dimLength : int the number of values allowed for this dimension in the array. | |
| sequence : int non negative integer that orders the dimensions in the array coordinate scheme | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) ArrayDimension | Target: characterizedBy (Class) DataElement Cardinality: [1] |

2.2.2 ArrayVariable

A data element that represents a multidimensional array of values of the same type (OGC 10-090r3). The dimension properties associated with the variable define the axes of the array. ArrayVariable.metaAttribute properties describe the gridding scheme used to assign values to the dimension coordinates for the array cells. Array variables are used to

represent a coverage (see ISO19123); this part of the model is not detailed here and should be treated as an extension to the metaAttribute class.

OUTGOING STRUCTURAL RELATIONSHIPS

Generalization from ArrayVariable to DataElement [Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

Aggregation from ArrayDimension to ArrayVariable [Direction is 'Destination -> Source'.]

2.2.3 Concept

A mental phenomena that human beings use in their internal representation of the world. Webster's dictionary [1996] uses the terms 'idea' and 'object of thought' to convey the meaning of 'concept.' Concepts exist in the mind of human observers. Representation includes a definition in natural language that explains the idea for people to understand; and a unique, machine-parseable identifier (operationally, it's a string...) for use by computer software. The item also includes labels (words, designations) that are used in natural language communication to signify the concept. These labels are language-localized, and ideally would be context localized to account for different community practice using the same language. A preferred label in at least one language is required. Finally, the concept item should include source information citing the intellectual origin of the definition. Concept register items are the base representations in the vocabulary registry for various classes that are extended with additional properties for the Information Model Registry. These classes include DataType (logical and syntactic/primitive), MeasureClass, ArrayDimension, ObjectClass, Property, and UnitOfMeasure.

CONSTRAINTS

Invariant. count (conceptURI) + count(definition) > 0
either a conceptURI or a definition SHALL be provided for each concept.

OUTGOING STRUCTURAL RELATIONSHIPS

Aggregation from Concept to ConceptScheme
[Direction is 'Source -> Destination'.]

INCOMING STRUCTURAL RELATIONSHIPS

Generalization from SyntacticDataType to Concept[Direction is 'Source -> Destination'.]
Generalization from LogicalDataType to Concept[Direction is 'Source -> Destination'.]
Generalization from MeasureClass to Concept[Direction is 'Source -> Destination'.]
Generalization from ArrayDimension to Concept[Direction is 'Source -> Destination'.]
Generalization from UnitOfMeasure to Concept[Direction is 'Source -> Destination'.]
Generalization from ObjectClass to Concept[Direction is 'Source -> Destination'.]
Generalization from Property to Concept[Direction is 'Source -> Destination'.]

ATTRIBUTES

conceptURI : anyURI Multiplicity: ([0..1],) a URI that identifies the concept.
prefName : text preferred name for humans to use when talking about this concept.
definition : text Multiplicity: ([0..1],)
notes : text Multiplicity: ([0..1],) non normative information about the derivation of the concept
altLabel : text@lang Multiplicity: ([0..*],) other language-localized text strings by which the concept may be known in other contexts.

ASSOCIATIONS

Association (direction: Source -> Destination)
Source: (Class) Concept Target: source (Class) Citation Cardinality: [1..*]

2.2.4 ConceptScheme

A collection of concepts defined within a single conceptSpace. The implication of definition in the concept space is that every concept in the scheme denotes a value or value range on each axis (base dimension) of the concept space.

| INCOMING STRUCTURAL RELATIONSHIPS | |
|--|---|
| Aggregation from Concept to ConceptScheme [Direction is 'Source -> Destination'.] | |
| ATTRIBUTES | |
| identifier : anyURI | |
| schemeName : text | |
| intendedUse : text Multiplicity: ([0..1],) | |
| conceptAxes : Concept Multiplicity: ([0..*],) if axes are specified, implication is that every concept that is a member of the scheme denotes some value or range of values for each axis. | |
| source : Citation Multiplicity: ([0..*],) | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) ConceptualDomain | Target: conceptSpace (Class) ConceptScheme Cardinality: [0..1] |

2.2.5 ConceptualDomain

a set of **value meanings** which may either be enumerated or expressed via a description. In ISO11179, *Conceptual_Domain* is an abstract class, which has two possible subclasses: *Enumerated_Conceptual_Domain* and *Described_Conceptual_Domain*. Every *Conceptual_Domain* instance must be either an *Enumerated_Conceptual_Domain* or a *Described_Conceptual_Domain* or a combination of the two. These subtypes are not modeled here.

| ATTRIBUTES | |
|--|---|
| identifier : anyURI | |
| domainName : text | |
| conceptualValueType : Concept high level categorization of the kind of values in this domain: e.g. narrative text, count, coordinate measurement, ratio measurement, interval measurement, concept, truth value, DateTime, Date, Time, vector, continuous field, sequence, name, rate (see 19103, maybe ISO80000?) | |
| description : text Multiplicity: ([0..1],) | |
| source : Citation Multiplicity: ([0..1],) | |
| ASSOCIATIONS | |
| Association (direction: Bi-Directional) Source: meaning (Class) ConceptualDomain Cardinality: [0..1] | Target: representation (Class) ValueDomain Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) ConceptualDomain | Target: applicableUnits (Class) MeasureClass Cardinality: [1..*] |
| Association (direction: Source -> Destination) Source: (Class) ConceptualDomain | Target: conceptSpace (Class) ConceptScheme Cardinality: [0..1] |
| Association (direction: Source -> Destination) Source: (Class) Property | Target: domain (Class) ConceptualDomain |

2.2.6 Constraint

constraints on the instances of the dataType in this data element--range, required units, other logical constraints.

| INCOMING STRUCTURAL RELATIONSHIPS | |
|---|--|
| Generalization from RangeConstraint to Constraint [Direction is 'Source -> Destination'.] | |
| ATTRIBUTES | |
| constraintStatement : text statement of the constraint | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) | |

| ASSOCIATIONS | |
|-----------------------------|--|
| Source: (Class) ValueDomain | Target: domainConstraint (Class) Constraint Cardinality: [0..*] |

2.2.7 ControlledVocabulary

A collection of terms used as the allowable values for an enumeratedValueDomain. Implements a concept scheme by specifying specific designations (strings) for each concept in the scheme.

| INCOMING STRUCTURAL RELATIONSHIPS |
|---|
| Aggregation from Term to ControlledVocabulary [Direction is 'Destination -> Source'.] |

| ATTRIBUTES |
|--|
| identifier : anyURI |
| vocabularyName : text |
| description : text Multiplicity: ([0..1],) |
| source : Citation Multiplicity: ([0..*],) |

| ASSOCIATIONS | |
|---|--|
| Association (direction: Source -> Destination) Source: (Class) EnumeratedValueDomain | Target: codelist (Class) ControlledVocabulary Cardinality: [0..*] role name from ISO19115 used here |
| Association (direction: Source -> Destination) Source: (Class) ImplementationElement | Target: implementedCodelist (Class) ControlledVocabulary Cardinality: [0..1] |

2.2.8 DataElement

An information object that represents a unit of data that quantifies a property in the context of an ObjectClass. The identity of a DataElement is based on its meaning and domain. The intention is that a DataElement does not denote a particular implementation environment, corresponding to 'logical model' data modeling approaches.

| OUTGOING STRUCTURAL RELATIONSHIPS |
|--|
| Aggregation from DataElement to DataObject [Direction is 'Destination -> Source'.] |

| INCOMING STRUCTURAL RELATIONSHIPS |
|--|
| Generalization from ArrayVariable to DataElement [Direction is 'Source -> Destination'.] |

| ATTRIBUTES |
|---|
| identifier : anyURI |
| elementName : text full name to designate this DataElement in the context of the containing DataObject. |
| prefLabel : text Multiplicity: ([0..1],) the label suggested for use to identify the DataElement in tables for computer use; generally a shorter version of the full elementName. |
| description : text Multiplicity: ([0..1],) documentation of any special considerations for the implementation of the dataElementConcept by this element. |
| source : Citation |

| ASSOCIATIONS | |
|--|---|
| Association (direction: Source -> Destination) Source: (Class) DataElement | Target: metaAttribute (Class) MetaAttribute Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) DataElement | Target: domain (Class) ValueDomain Cardinality: [1] |
| Association (direction: Source -> Destination) Source: logicalModel (Class) DataElement | Target: implementation (Class) ImplementationElement Cardinality: [0..*] |
| Association (direction: Source -> Destination) | |

| ASSOCIATIONS | |
|--|--|
| Source: representation (Class) DataElement Cardinality: [0..*] | Target: meaning (Class) Property Cardinality: [1] |
| Association (direction: Source -> Destination) | |
| Source: (Class) ArrayDimension | Target: characterizedBy (Class) DataElement Cardinality: [1] |

2.2.9 DataMetaAttributes

Example attributes that might be associated with a DataObject. From ISO11179

| ATTRIBUTES |
|--|
| samplingBasisURI : anyURI Multiplicity: ([0..1],) |
| samplingBasisName : text Multiplicity: ([0..1],) name of the entity (samplingFeature) that is the basis for assigning property values to the feature of interest. Examples include borehole interval, ground sampling cell (for a remote sensing image), outcrop station, physical sample, outcrop area, instrument, aggregation (of other data). The spatial representation section describes how the sampling features are geolocated |
| samplingBasisDescription : text Multiplicity: ([0..1],) |
| dElemGroupURI : anyURI Multiplicity: ([0..1],) document a group definition as a separate attribute, requires an assigned attributeURI for the group, then use that URI here to aggregate attributes in a group. The GroupURI may be externally defined, but the URI must dereference to provide an explanation of the group |
| geometryTypeName : text Multiplicity: ([0..1],) term that specifies the type of geometry represented by this property, should be linked to a controlled vocabulary. Example (ISO19107 geometryType codelist): curve, line, point, polygon, solid, surface, grid. Should distinguish 2D and 3D point |
| geometryTypeURI : anyURI |
| geometryTypeVocabularyURI : anyURI Multiplicity: ([0..1],) Unique identifier for the vocabulary within which the geometry type is identified; if the geometryTypeURI identifier type allows resolution of the vocabulary, this is redundant; for some schemes the TypeURI value may be a token that is appended to the vocabulary URI to obtain the full TypeURI |
| discreteSamplingSchemaType : text Multiplicity: ([0..1],) Term or identifier that categorizes the schema used to describe the sampling structure for a gridded/array type data (discrete coverage), using a short string for machine processing. Typically the various dimensions map to some conceptual domain like space or time, sampled at some interval, See ISO19123, ISO19115-1 and 19115-2 MD_CoverageDescription and MD_GridSpatialRepresentation |
| discreteSamplingExplanation : text Multiplicity: ([0..1],) explanation of how the discrete coverage is represented and what kind of content is encoded here, for human reader. Actual discreteSamplingDescription is unconstrained JSON blob included following this element. |
| discreteSamplingDescription : JSONObject Multiplicity: ([0..1],) JSON content that describes the discrete sampling geometry, according to the description type specified by discreteSamplingDescriptionType |

2.2.10 DataObject

An information object that represents an entity of interest (ObjectClass in this model, based on ISO11179) in some domain; the representation consists of a collection of DataElements that are used to quantify properties of instances of the entity. Corresponds to 'dataType' in ISO11179, Entity in Entity-Relationship models, Object in object models, 'document' in document type noSQL databases (e.g. CouchDb, MongoDB), 'Variable' in the netCDF common data model (OGC 10-090r3).

| INCOMING STRUCTURAL RELATIONSHIPS |
|--|
| Aggregation from DataElement to DataObject [Direction is 'Destination -> Source'.] |

| ATTRIBUTES |
|--|
| identifier : anyURI |
| label : text Label by which the data object is identified in its application context |
| logicType : Concept categorize the logical paradigm for the representation-- e.g. relational, object-oriented, graph, tabular text |
| expectedUse : text Multiplicity: ([0..*],) an explanation of the intention for the dataType Properties: source = RDA DataType Model 2015 |

| ASSOCIATIONS | |
|--|---|
| Association (direction: Source -> Destination) Source: (Class) DataObject | Target: examples (Class) Example Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) DataObject | Target: typeDefinitionProvenance (Class) ProcessingStep Cardinality: [0..*] |

| ASSOCIATIONS | |
|--|---|
| Association (direction: Source -> Destination) Source: (Class) DataObject | Target: metaAttribute (Class) MetaAttribute Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: logicalModel (Class) DataObject | Target: implementation (Class) ImplementationObject Cardinality: [0..*] |
| Association (direction: Bi-Directional) Source: meaning (Class) ObjectClass Cardinality: [1] | Target: representation (Class) DataObject Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) ObjectDataType | Target: schema (Class) DataObject Cardinality: [1] |

2.2.11 EnumeratedValueDomain

| CONSTRAINTS |
|--|
| Invariant. type (dataType) = LogicalDataType the values provided by an EnumeratedValueDomain are text strings, thus the dataType for the valueDomain SHALL not be an ObjectDataType [Approved, Weight is 0.] |

| OUTGOING STRUCTURAL RELATIONSHIPS |
|--|
| Generalization from EnumeratedValueDomain to ValueDomain [Direction is 'Source -> Destination'.] |

| ASSOCIATIONS | |
|---|--|
| Association (direction: Source -> Destination) Source: (Class) EnumeratedValueDomain | Target: codelist (Class) ControlledVocabulary Cardinality: [0..*] role name from ISO19115 used here |

2.2.12 Example

Implementation of DataType.example from RDA dataType model 2015

| ATTRIBUTES |
|--|
| exampleDataTypeInstance : text Multiplicity: ([0..1],) a text encoding of an example instance of the data type |
| linkToExampleDataset : Link Multiplicity: ([0..1],) Link to an example dataset that instantiates the data type |

| ASSOCIATIONS | |
|--|---|
| Association (direction: Source -> Destination) Source: (Class) DataObject | Target: examples (Class) Example Cardinality: [0..*] |

2.2.13 ImplementationElement

representation of the physical implementation of a DataElement. The targetEnvironment and purpose are inferred from the containing ImplementationObject

| ATTRIBUTES |
|--|
| identifier : anyURI |
| localName : text |
| cardinality : text |
| sequence : int Multiplicity: ([0..1],) if the order of the attributes in the implementation instance is fixed, sequence numbers SHALL be provided to define the order. |
| alias : text@lang Multiplicity: ([0..*],) other names that may be used to identify the dataElement; should be language or context-localized. |

| ASSOCIATIONS |
|--|
| Association (direction: Source -> Destination) |

| ASSOCIATIONS | |
|--|---|
| Source: (Class) ImplementationElement | Target: implementedCodelist (Class) ControlledVocabulary Cardinality: [0..1] |
| Association (direction: Source -> Destination) Source: (Class) ImplementationElement | Target: implementationDataType (Class) SyntacticDataType Cardinality: [1] |
| Association (direction: Source -> Destination) Source: (Class) ImplementationObject | Target: implementedAttribute (Class) ImplementationElement Cardinality: [1..*] |
| Association (direction: Source -> Destination) Source: logicalModel (Class) DataElement | Target: implementation (Class) ImplementationElement Cardinality: [0..*] |

2.2.14 ImplementationObject

representation of a physical implementation of a DataObject.

| ATTRIBUTES |
|---|
| identifier : anyURI |
| localName : text designation for this information object in its native environment |
| description : text Multiplicity: ([0..1],) |
| targetEnvironment : Concept Multiplicity: ([1..*],) identification of the specific software environment for which this implementation is designed. e.g. Oracle 10 relational db, XML v1.0, GML 3.2 application schema |
| purpose : Concept Multiplicity: ([0..*],) categorize the intention of this implementation, e.g. interchange format, database table, data acquisition tool, data archive, object oriented software, semantic application |

| ASSOCIATIONS | |
|---|---|
| Association (direction: Source -> Destination) Source: (Class) ImplementationObject | Target: implementedAttribute (Class) ImplementationElement Cardinality: [1..*] |
| Association (direction: Source -> Destination) Source: (Class) InterchangeFormat | Target: encodes (Class) ImplementationObject Cardinality: [1..*] |
| Association (direction: Source -> Destination) Source: logicalModel (Class) DataObject | Target: implementation (Class) ImplementationObject Cardinality: [0..*] |

2.2.15 InterchangeFormat

a document type definition used to serialize information for one or more ImplementationObjects

| ATTRIBUTES |
|---|
| identifier : anyURI |
| formatName : text |
| label : text Multiplicity: ([0..1],) short text version of format name. In data that is identifying this interchange format using a text string (not a URI), this string should be used (if specified). |
| description : text Multiplicity: ([0..1],) |
| fileType : Concept mime type for the file |
| source : Citation should identify an accessible document that defines the interchange format. |
| schemaDocument : Link Multiplicity: ([0..1],) a link to a schema document (xsd, schematron, RuleML) that can be used to validate instance documents. |
| schemaType : Concept Multiplicity: ([0..1],) category that identifies the kind of schema document available to validate interchange document instances. |

| ASSOCIATIONS | |
|---|---|
| Association (direction: Source -> Destination) Source: (Class) InterchangeFormat | Target: encodes (Class) ImplementationObject Cardinality: [1..*] |

2.2.16 LogicalDataType

A set of distinct values, characterized by properties of those values and by operations on those values. E.g. string, integer, real number, term, reference. Implementation independent, but restricts implementation options. Must be consistent with *ConceptualValueType* of meaning. *ConceptualDomain* (if one is specified).

Does not have *UnitOfMeasure* property; measure is considered an object that has a numeric value and a UOM concept.

EXAMPLE 1

name: integer

description: mathematical datatype comprising the exact integral values.

scheme_reference: ISO/IEC 11404:2007

EXAMPLE 2

name: BL

description: BL stands for the values of two-valued logic. A BL value can be either true or false, or may have a null flavor.

scheme_reference: ISO 21090:2010

| OUTGOING STRUCTURAL RELATIONSHIPS | |
|---|--|
| Generalization from LogicalDataType to Concept [Direction is 'Source -> Destination'.] | |
| INCOMING STRUCTURAL RELATIONSHIPS | |
| Generalization from ObjectDataType to LogicalDataType [Direction is 'Source -> Destination'.] | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) ValueDomain | Target: dataType (Class) LogicalDataType |

2.2.17 MeasureClass

a set of equivalent **units of measure** that may be shared across multiple **dimensionalities**. *Measure_Class* allows a grouping of units of measure to be specified once, and reused by multiple dimensionalities.

EXAMPLE: We could define the *Measure_Classes*: Metric Linear Distance, Imperial Linear Distance, each associated with the appropriate *Units_of_Measure*; and associate them with *Dimensionalities*: Height, Width, and Depth to model the three spatial dimensions.

| OUTGOING STRUCTURAL RELATIONSHIPS | |
|--|---|
| Generalization from MeasureClass to Concept [Direction is 'Source -> Destination'.] | |
| INCOMING STRUCTURAL RELATIONSHIPS | |
| Aggregation from UnitOfMeasure to MeasureClass [Direction is 'Destination -> Source'.] | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) ConceptualDomain | Target: applicableUnits (Class) MeasureClass Cardinality: [1..*] |

2.2.18 MetaAttribute

Metadata attributes that document the data element intention and usage. NetCDF uses this to assign units to the attribute. DataObject (data type) attributes might include geometryType, samplingBasis,

a data element might have discrete sampling scheme information, units of measure, etc.

model element based on NetCDF common data model 'attribute' concept on dataset and on variable, and properties on DataType and DataElement in ISO11179

| ATTRIBUTES |
|--|
| key : DataElement this model allows metaAttributes to themselves be DataObjects or SyntacticDataTypes, which is more general than NetCDF CDM (OGC 10-090r3). Constraints: count(key.DataElement.metaAttribute)=0 : |
| values : text Multiplicity: ([1..*],) an array of 1 to many values assigned on this attribute. This is inherited from OGC10- |

| ATTRIBUTES | |
|--|--|
| 090r3; needs testing to determine if necessary. text data type is assigned, non-text values must be text encoded | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) DataElement | Target: metaAttribute (Class) MetaAttribute Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) DataObject | Target: metaAttribute (Class) MetaAttribute Cardinality: [0..*] |

2.2.19 ObjectClass

object class is a **concept** (3.2.18) that represents a set of ideas, abstractions, or things in the real world that can be identified with explicit boundaries and meaning and whose properties and behavior follow the same rules.

| OUTGOING STRUCTURAL RELATIONSHIPS | |
|---|--|
| Generalization from ObjectClass to Concept [Direction is 'Source -> Destination'.] | |
| INCOMING STRUCTURAL RELATIONSHIPS | |
| Aggregation from Property to ObjectClass [Direction is 'Destination -> Source'.] | |
| ASSOCIATIONS | |
| Association (direction: Bi-Directional) Source: meaning (Class) ObjectClass Cardinality: [1] | Target: representation (Class) DataObject Cardinality: [0..*] |

2.2.20 ObjectDataType

| OUTGOING STRUCTURAL RELATIONSHIPS | |
|---|--|
| Generalization from ObjectDataType to LogicalDataType [Direction is 'Source -> Destination'.] | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) ObjectDataType | Target: schema (Class) DataObject Cardinality: [1] |

2.2.21 ProcessingStep

implementation of RDA DataType.provenance

| ATTRIBUTES | |
|--|---|
| event : Event | |
| source : Citation Private Multiplicity: ([0..*],) | |
| contributor : QualifiedAttribution Multiplicity: ([1..*],) | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) DataObject | Target: typeDefinitionProvenance (Class) ProcessingStep Cardinality: [0..*] |

2.2.22 Property

A conceptual property. Implementation specifics included in the data type definition that uses this property
ISO11179: a quality common to all members of an **object class**. A property may be any feature that humans naturally use to distinguish one individual object from another. It is the human perception of a single quality of an object class in the

real world. It is conceptual and thus has no particular associated means of representation by which the property can be communicated.

A quality that inheres in an entity.

This is derived from ISO11179 **data element concept**: a **concept** that is an **association** of a **property** with an **object class**. A data element concept can be represented in the form of a **data element**, described independently of any particular representation. Since elementProperty is mandatory and single valued, there doesn't seem to be much gained by separating property and dataElementConcept

| OUTGOING STRUCTURAL RELATIONSHIPS |
|--|
| Aggregation from Property to ObjectClass [Direction is 'Destination -> Source'.] |
| Generalization from Property to Concept [Direction is 'Source -> Destination'.] |

| ATTRIBUTES |
|--|
| <p>dimensionality : Concept Multiplicity: ([0..1],) An expression of measurement without units. Each dimension category groups a set equivalent units of measure, where equivalence is determined by the existence of a quantity-preserving one-to-one correspondence between values measured in one unit of measure and values measured in the other unit of measure, independent of context, and where the characterizing operations are the same. Appears to correspond (exactMatch?) to NetCDF common data model 'dimension' concept: "represents a real physical dimension, for example, time, latitude, longitude, or height. A dimension might also be used to index other quantities, for example station or model-run-number." (NetCDF User Guide, Version 4.1.3, 2011-06). ISO/IEC 11179 also permits non-physical dimensions (e.g. value dimensions such as: currency, quality indicator).</p> <p>See also ISO 80000 (https://en.wikipedia.org/wiki/International_System_of_Quantities) for another specification of physical dimensions (e.g. length, mass, velocity). When a dimensionality is specified, then the Unit_of_Measure specified for any Value_Domain that is based on this Conceptual_Domain SHALL be consistent with this dimensionality.</p> <p>EXAMPLES: inches, feet, meters, and centimeters are all units of measure whose dimensionality is length. Other common dimensionalities include: mass, time, area, volume, etc.</p> <p>NOTES: -- The units of measure "temperature in degrees Fahrenheit" and "temperature in degrees Celsius" have the same dimensionality, because given a value measured in degrees Fahrenheit there is a value measured in degrees Celsius that is the same quantity, and vice-versa. This assumes we are dealing with temperature coordinates. There is no offset when converting among temperature interval measures, e.g., the temperature difference between the coldest and hottest temperature on a day. -- Quantities may be grouped together into categories of quantities which are mutually comparable. Lengths, diameters, distances, heights, wavelengths and so on would constitute such a category. Mutually comparable quantities have the same dimensionality if they have common characterizing operations. The requirement of common "characterizing operations" for all units of measure with the same dimensionality is a stronger requirement than that commonly adopted in conventional dimensional analysis (where comparability and transformability usually suffice). Thus with respect to temperature, absolute temperature coordinates (e.g. Kelvins) are here considered to be a different dimensionality than "offset" temperature coordinates (e.g. degrees Celsius or Fahrenheit). It is meaningful to take the ratio of absolute temperature coordinates, but not of "offset" temperature coordinates, wherein the arbitrary translation of zero renders ratios meaningless. The notion of characterizing operations used here has been adapted from the statistics literature where distinctions are commonly made among categorical, ordered, interval, and ratio measures. (ISO11179)</p> |

| ASSOCIATIONS | |
|--|---|
| Association (direction: Source -> Destination) Source: (Class) Property | Target: domain (Class) ConceptualDomain |
| Association (direction: Source -> Destination) Source: representation (Class) DataElement Cardinality: [0..*] | Target: meaning (Class) Property Cardinality: [1] |

2.2.23 RangeConstraint

| OUTGOING STRUCTURAL RELATIONSHIPS |
|---|
| Generalization from RangeConstraint to Constraint [Direction is 'Source -> Destination'.] |

| ATTRIBUTES |
|-------------------|
| minValue : number |
| maxValue : number |

2.2.24 SyntacticDataType

physical data type, specifying string length limits, specific number implementation (long integer, float, double), list syntax, symbols to use for boolean values, etc..

| OUTGOING STRUCTURAL RELATIONSHIPS | |
|---|--|
| Generalization from SyntacticDataType to Concept [Direction is 'Source -> Destination'.] | |
| ATTRIBUTES | |
| format : text Multiplicity: ([0..1],) template for the structure of the presentation of the value(s) EXAMPLE – YYYY-MM-DD for a date., limitations on character string length. Typically some sort of regular expressions specifying syntax for the alphanumeric string that specifies data values. If the data type allows lists, format text should specify list boundary and delimiter characters. | |
| annotation : text Multiplicity: ([0..1],) specifying information to further define the Datatype | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) ImplementationElement | Target: implementationDataType (Class) SyntacticDataType Cardinality: [1] |

2.2.25 Term

a representation of a concept in a particular application

| OUTGOING STRUCTURAL RELATIONSHIPS | |
|---|--|
| Aggregation from Term to ControlledVocabulary [Direction is 'Destination -> Source'.] | |
| ATTRIBUTES | |
| designation : text String used to represent a concept in this application | |
| meaning : Concept Concept that denotes meeting of the value in this application | |

2.2.26 UnitOfMeasure

the units in which associated values are measured.

Units of measure are not limited to physical categories. Examples of physical categories are: linear measure, area, volume, mass, velocity, time duration. Examples of non-physical categories are: currency, quality indicator, color intensity.

| OUTGOING STRUCTURAL RELATIONSHIPS | |
|--|--|
| Aggregation from UnitOfMeasure to MeasureClass [Direction is 'Destination -> Source'.] | |
| Generalization from UnitOfMeasure to Concept [Direction is 'Source -> Destination'.] | |

2.2.27 ValueDomain

a set of **permissible values**. A *Value_Domain* provides representation, but has no implication as to what *Data_Element_Concept* the values are associated with, nor what the values mean. A *Value_Domain* may be associated with multiple *Data_Elements*.

| INCOMING STRUCTURAL RELATIONSHIPS | |
|--|--|
| Generalization from EnumeratedValueDomain to ValueDomain [Direction is 'Source -> Destination'.] | |
| ATTRIBUTES | |
| identifier : anyURI | |
| domainName : text | |
| description : text Multiplicity: ([0..1],) description of intention of domain | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) ValueDomain | Target: domainConstraint (Class) Constraint Cardinality: [0..*] |
| Association (direction: Source -> Destination) | |

| ASSOCIATIONS | |
|---|---|
| Source: (Class) ValueDomain | Target: dataType (Class) LogicalDataType |
| Association (direction: Source -> Destination) Source: (Class) DataElement | Target: domain (Class) ValueDomain Cardinality: [1] |
| Association (direction: Bi-Directional) Source: meaning (Class) ConceptualDomain Cardinality: [0..1] | Target: representation (Class) ValueDomain Cardinality: [0..*] |

3 HelperClasses

Citation, Event, Agent, Qualified attribution, address. Various metadata classes useful for other model attribute values.

3.1 Diagrams

3.1.1 Agent-Event-Link diagram

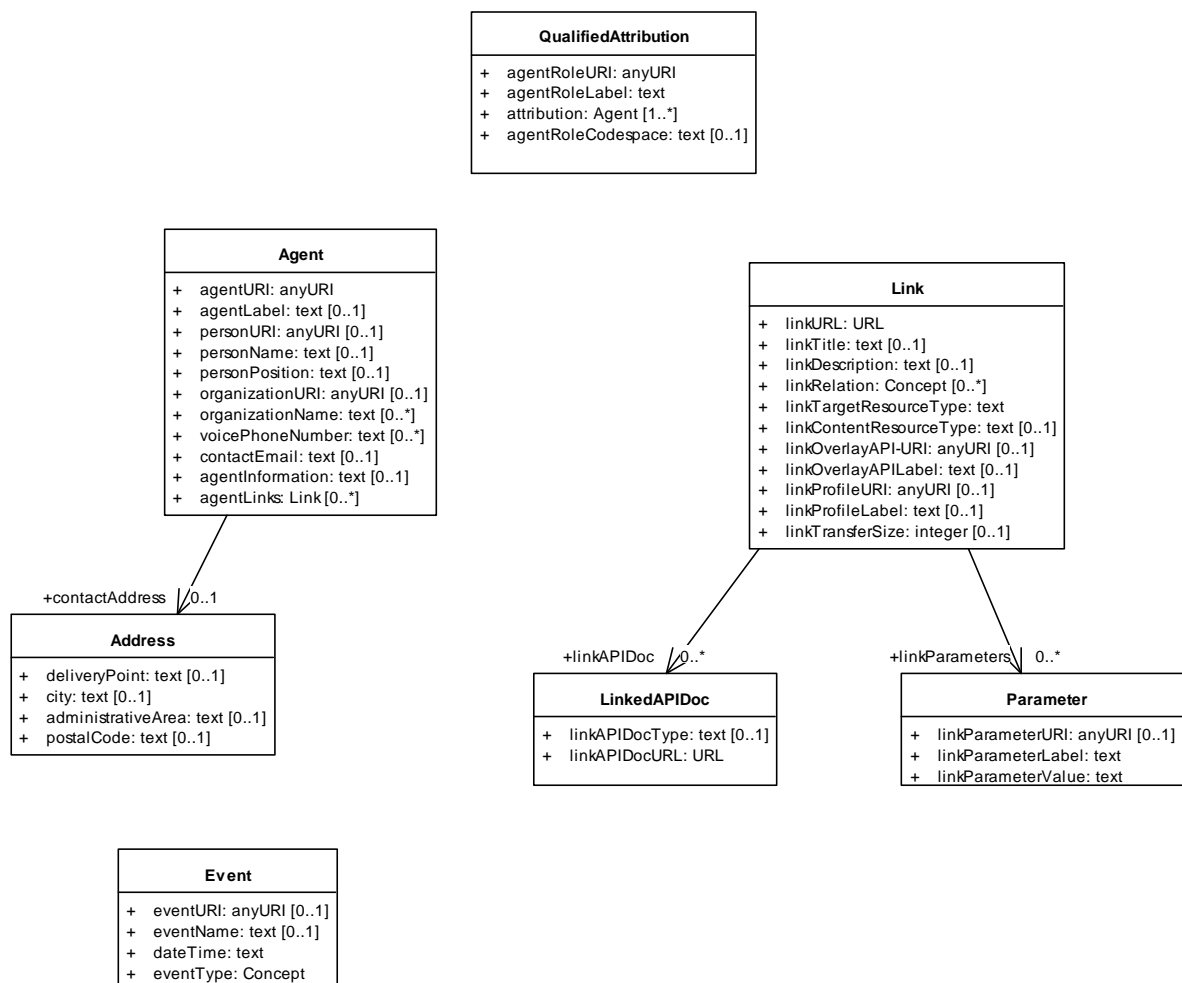


Figure 10: AgentEventLink models

3.1.2 Citation diagram

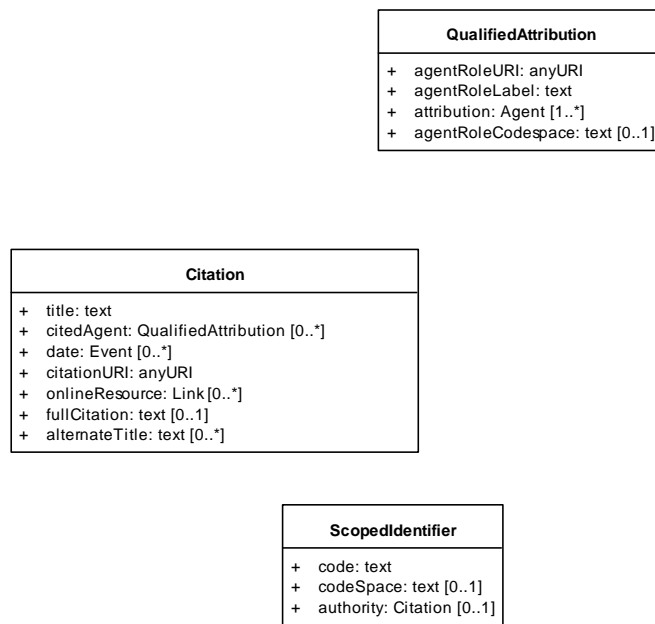


Figure 11: Citation

3.2 Classes

3.2.1 Address

| ATTRIBUTES | |
|---|---|
| deliveryPoint : text Multiplicity: ([0..1],) | street address, PO box. Full postal address should be in this field if it is not parsed into separate fields. |
| city : text Multiplicity: ([0..1],) | |
| administrativeArea : text Multiplicity: ([0..1],) | |
| postalCode : text Multiplicity: ([0..1],) | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) Agent | Target: contactAddress (Class) Address Cardinality: [0..1] |

3.2.2 Agent

"An agent is something that bears some form of responsibility for an activity taking place, for the existence of an entity, or for another agent's activity." (W3C PROV). In this view, software can also be an agent. Responsibility does not have to be 'conscious' or intentional. An agent is an identifiable entity; could be an organization, an individual who may or may not be associated with an organization, an entity identified via a role (position) relative to an organization, or an artificial entity (software, a machine).

| CONSTRAINTS |
|---|
| Invariant. count(personName + personPosition+organizationName) >0 at least one text name SHALL be provided |
| Invariant. count(voicePhoneNumber + contactEmail) > 0 at least one voice phone number or e-mail SHALL be provided |
| ATTRIBUTES |
| agentURI : anyURI Unique identifier for the agent. At least follow URI syntax (e.g. a prefix for the 'protocol'), even if they are local identifiers. |
| agentLabel : text Multiplicity: ([0..1],)text string to identify the agent for humans. |
| personURI : anyURI Multiplicity: ([0..1],)unique identifier for a person. |
| personName : text Multiplicity: ([0..1],)The name of an individual. Suggested format--Last Name,First name MI |
| personPosition : text Multiplicity: ([0..1],)identifies an individual who currently holds the named position in context of an |

| ATTRIBUTES |
|---|
| organization |
| organizationURI : anyURI Multiplicity: ([0..1],) |
| organizationName : text Multiplicity: ([0..*],)unique identifier for organization; use to link to the organization that the person is affiliated with, in the case that the AgentURI is for the person and they have an organization affiliation. In this case the organizationName is equivalent to the linkLabel property |
| voicePhoneNumber : text Multiplicity: ([0..*],)Number for voice contact. Use registered tel URI scheme for encoding. See http://tools.ietf.org/html/rfc3966 |
| contactEmail : text Multiplicity: ([0..1],) |
| agentInformation : text Multiplicity: ([0..1],)text information, providing other details useful for making contact, like hours or ordering instructions |
| agentLinks : Link Multiplicity: ([0..*],) |

| ASSOCIATIONS |
|--|
| Association (direction: Source -> Destination) Source: (Class) Agent Target: contactAddress (Class) Address Cardinality: [0..1] |

3.2.3 Citation

a resolvable reference to an information source. Not detailed here, Should include a text string providing guidance on how to cite the source, a title, bibliographic information if appropriate, a URI for the source resource, and Links to access representations of the resource online, and optionally a relation concept explaining the relationship between the source and the citing element.

| ATTRIBUTES |
|--|
| title : text |
| citedAgent : QualifiedAttribution Multiplicity: ([0..*],)implement CI_Responsibility using QualifiedAttribution based on PROV |
| date : Event Multiplicity: ([0..*],) |
| citationURI : anyURI URIs, ISBN, ISSN, other alternate identifiers for the cited resource. |
| onlineResource : Link Multiplicity: ([0..*],)implements ISO19115 onlineResource and graphic properties of CI_Citation. Browse graphic is implemented as a link to an online graphic. |
| fullCitation : text Multiplicity: ([0..1],)suggested citation text to use in referencing or citing this resource in text. This property includes edition, editionDate, series, otherCitationDetails from ISO19115-1 CI_Citation. |
| alternateTitle : text Multiplicity: ([0..*],) |

| ASSOCIATIONS |
|---|
| Association (direction: Source -> Destination) Source: (Class) Concept Target: source (Class) Citation Cardinality: [1..*] |

3.2.4 Event

Associates an event defined by an eventType and an optional event instance URI, with a date time string detailing when the event occurred. In simplest case can just provide dateTime if the event context is unambiguous in the importing object.

| ATTRIBUTES |
|---|
| eventURI : anyURI Multiplicity: ([0..1],)identifier for the event instance if this object is about a specific event |
| eventName : text Multiplicity: ([0..1],)name to identify the event for people |
| dateTime : text Use ISO 8601: 2011-10-11T14:30 |
| eventType : Concept |

3.2.5 Link

| ATTRIBUTES |
|---|
| linkURL : URL a web-derferencable identifier that locates the link target; typically an HTTP URI. URI syntax specifies that the identifier string includes a prefix that specifies the base protocol for the identifier |
| linkTitle : text Multiplicity: ([0..1],) |

| ATTRIBUTES | |
|--------------------------------|---|
| linkDescription : text | Multiplicity: ([0..1],) free text description of the target to help UI |
| linkRelation : Concept | Multiplicity: ([0..*],) |
| linkTargetResourceType : text | typically use MIME type string from IANA registry http : //www.iana.org/assignments/media-types/application/index.html . This is the type of the file that will be accessed directly by the link URL; if this is a container file (e.g. zip archive), the innerResourceType property is used to specify the type of file with actual resource content. |
| linkContentResourceType : text | Multiplicity: ([0..1],) File type for the actual resource content. Typically use MIME type string from IANA registry http : //www.iana.org/assignments/media-types/application/index.html |
| linkOverlayAPI-URI : anyURI | Multiplicity: ([0..1],) Identifier for overlay API. Reference that identifies the API for messages tunneled to a component on the target server. Optional, provide if such scheme or protocol is necessary to utilize the link. The URI should be defined by the service specification for the protocol or service type; version information should be included if applicable. E.g. OGC WMS, WS-services. This property is typically used for services that encode remote procedure calls using identifiers dereferenced using standard HTTP methods (GET, POST). |
| linkOverlayAPILabel : text | Multiplicity: ([0..1],) Text to identify the overlay API in a UI. Reference that identifies the API for messages tunneled to a component on the target server. |
| linkProfileURI : anyURI | Multiplicity: ([0..1],) |
| linkProfileLabel : text | Multiplicity: ([0..1],) Text to identify the profile in a UI. Identifier for profile of specifications identified by type, overlayAPI, and template attributes. Optional, provide if additional conventions are necessary for content contained in messages through this link. Note that the same output scheme might be encoded using different types. Profiles typically add usage conventions when the interchange scheme offers alternate approaches, restrict cardinality for elements in the interchange format, or specify usage of particular vocabularies. |
| linkTransferSize : integer | Multiplicity: ([0..1],) advisory length of the linked content in octets |

| ASSOCIATIONS | |
|--|---|
| Association (direction: Source -> Destination) Source: (Class) Link | Target: linkAPIDoc (Class) LinkedAPIDoc Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) Link | Target: linkParameters (Class) Parameter Cardinality: [0..*] |

3.2.6 LinkedAPIDoc

URLs that will get descriptions of the link operation; particularly targeted for RESTful type links. API doc should describe the resource architecture of the endpoint and provide some example requests, maybe even a template. Each URL is scoped to an API document type, e.g. Swagger, Hydra, Siren, HAL...

| ATTRIBUTES | |
|-----------------------|--|
| linkAPIDocType : text | Multiplicity: ([0..1],) text string that identifies the API doc type; ideally a URI defined in the API doc specification |
| linkAPIDocURL : URL | URL that will get the API doc describing operation of the link |

| ASSOCIATIONS | |
|--|--|
| Association (direction: Source -> Destination) Source: (Class) Link | Target: linkAPIDoc (Class) LinkedAPIDoc Cardinality: [0..*] |

3.2.7 Parameter

key-value pair specifies parameters name and value, or properties that need to be associated with the link, e.g. WFS feature typeNames, WMS layer names

| ATTRIBUTES | |
|---------------------------|---|
| linkParameterURI : anyURI | Multiplicity: ([0..1],) identifier for the parameter from a controlled vocabulary |
| linkParameterLabel : text | string to display, and identifier to link parameter to registry |
| linkParameterValue : text | |

| ASSOCIATIONS | |
|--|---|
| Association (direction: Source -> Destination) Source: (Class) Link | Target: linkParameters (Class) Parameter Cardinality: [0..*] |

3.2.8 QualifiedAttribution

CI_ResponsibleParty, CI_Responsibility in ISO19115 or 19115-1. binds an agent to a role. Name if from W3C PROV

| ATTRIBUTES |
|--|
| agentRoleURI : anyURI identifier for role concept. Use Null URI if not available. If no URI is reported, then a codespace value should be considered mandatory |
| agentRoleLabel : text |
| attribution : Agent Multiplicity: ([1..*],) |
| agentRoleCodespace : text Multiplicity: ([0..1],) identifier for the vocabulary or codespace within which the role term is defined |

3.2.9 ScopedIdentifier

Implements MD_Identifier from ISO19115-1.

| ATTRIBUTES |
|---|
| code : text the identifier string value. If a codeSpace is provided, then the concatenation of the codeSpace and code SHALL be a globally unique identifier string. |
| codeSpace : text Multiplicity: ([0..1],) |
| authority : Citation Multiplicity: ([0..1],) |

4 RDA DataType model

Based on RDA Data Type Registries Working Group Output, (March 2015). UML representation is interpretation of the WG output by S.M. Richard (2015-08)

4.1 DataTypeModel diagram

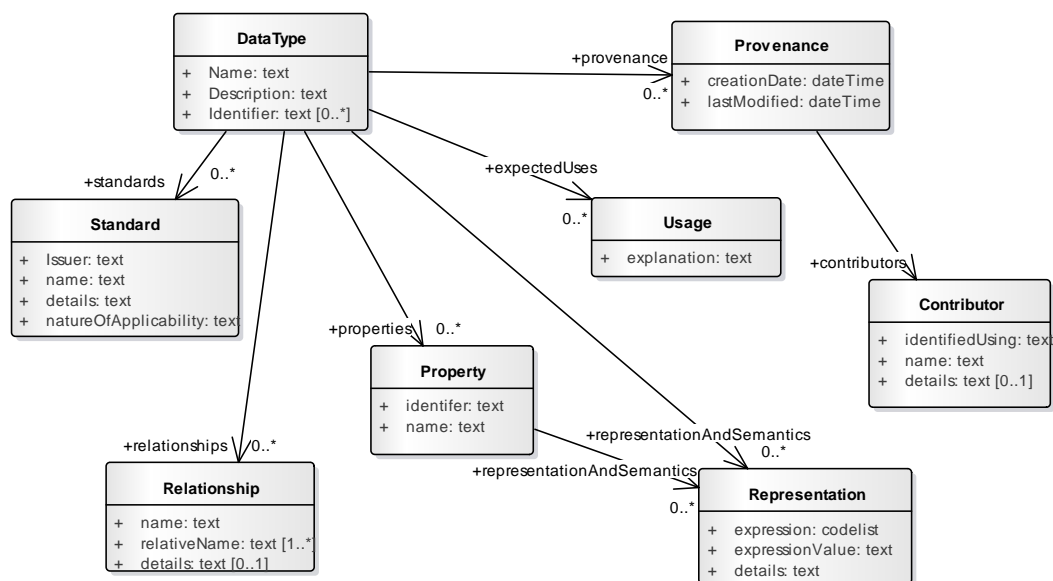


Figure 12: DataTypeModel

4.2 Contributor

| ATTRIBUTES | |
|--|---|
| identifiedUsing: (text) | |
| name: (text) | |
| details: (text). Multiplicity: ([0..1]) | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) Provenance | Target: contributors (Class) Contributor Contributors of this Record |

4.3 DataType

| ATTRIBUTES | |
|--|---|
| Name: (text) | |
| Description: (text) | |
| Identifier: (text). Multiplicity: ([0..*]) | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: expectedUses (Class) Usage Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: properties (Class) Property Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: representationAndSemantics (Class) Representation Cardinality: [0..*]this is soft typed |

| ASSOCIATIONS | |
|--|--|
| | with { expression, value } pairs |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: provenance (Class) Provenance Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: standards (Class) Standard Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: relationships (Class) Relationship Cardinality: [0..*] Intent: How the properties are related to each other, e.g., grouping of properties, cardinality, etc., should be captured here. |

4.4 Property

| ATTRIBUTES | |
|--|---|
| identifier: (text). TID of Existing Data Type | |
| name: (text). Name assigned to dependent type in this context | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) Property | Target: representationAndSemantics (Class) Representation Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: properties (Class) Property Cardinality: [0..*] |

4.5 Provenance

| ATTRIBUTES | |
|--|--|
| creationDate: (dateTime) | |
| lastModified: (dateTime) | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) Provenance | Target: contributors (Class) Contributor Contributors of this Record |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: provenance (Class) Provenance Cardinality: [0..*] |

4.6 Relationship

| ATTRIBUTES | |
|--|--|
| name: (text) | |
| relativeName: (text). Multiplicity: ([1..*]) | |
| details: (text). Multiplicity: ([0..1]) | |
| ASSOCIATIONS | |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: relationships (Class) Relationship Cardinality: [0..*] Intent: How the properties are related to each other, e.g., grouping of properties, cardinality, etc., should be captured here. |

4.7 Representation

Representation and Semantic Expression

| ATTRIBUTES |
|--|
| expression: (codelist). "suggestedVocabulary": ["Format", "Character Set", "Encoding", "Measurement Unit"] |
| expressionValue: (text). Unicode, UTF-8, Meter, etc. |
| details: (text) |

| ASSOCIATIONS | |
|--|--|
| Association (direction: Source -> Destination) Source: (Class) Property | Target: representationAndSemantics (Class) Representation Cardinality: [0..*] |
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: representationAndSemantics (Class) Representation Cardinality: [0..*]this is soft typed with {expression, value} pairs |

4.8 Standard

| ATTRIBUTES |
|--------------------------------|
| Issuer: (text) |
| name: (text) |
| details: (text) |
| natureOfApplicability: (text) |

| ASSOCIATIONS | | | |
|--|--|------------------------------------|--------------|
| Association (direction: Source -> Destination) | | Target: standards (Class) Standard | Cardinality: |
| Source: (Class) DataType | | | [0..*] |

4.9 Usage

| ATTRIBUTES |
|----------------------|
| explanation: (text) |

| ASSOCIATIONS | |
|--|---|
| Association (direction: Source -> Destination) Source: (Class) DataType | Target: expectedUses (Class) Usage Cardinality: [0..*] |