July 2022

Commons Ontology Library

Version 1.0 – beta 1

OMG Document Number: **ptc/22-07-01**

Normative reference: **https://www.omg.org/spec/Commons/1.0/**

This OMG document replaces the submission document (ad/22-06-01). It is an OMG Adopted Beta Specification and is currently in the finalization phase. Comments on the content of this document are welcome and should be directed to issues@omg.org by August 26, 2022.

You may view the pending issues for this specification from the OMG revision issues web page https://issues.omg.org/issues/lists.

The FTF Recommendation and Report for this specification will be published in December 2022. If you are reading this after that date, please download the available specification from the OMG Specifications Catalog.

USE OF SPECIFICATION – TERMS, CONDITIONS & NOTICES

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

LICENSES

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

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

PATENTS

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

GENERAL USE RESTRICTIONS

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

OMG's Issue Reporting Procedure

All OMG specifications are subject to continuous review and improvement. As part of this process we encourage readers to report any ambiguities, inconsistencies, or inaccuracies they may find by completing the Issue Reporting Form listed on the main web page https://www.omg.org, under Documents, Report a Bug/Issue (https://issues.omg.org/issues/create-new-issue).

Table of Contents

Preface

OMG

Founded in 1989, the Object Management Group, Inc. (OMG) is an open membership, not-for-profit computer industry standards consortium that produces and maintains computer industry specifications for interoperable, portable, and reusable enterprise applications in distributed, heterogeneous environments. Membership includes Information Technology vendors, end users, government agencies, and academia.

OMG member companies write, adopt, and maintain its specifications following a mature, open process. OMG's specifications implement the Model Driven Architecture® (MDA®), maximizing ROI through a full-lifecycle approach to enterprise integration that covers multiple operating systems, programming languages, middleware and networking infrastructures, and software development environments. OMG's specifications include: UML® (Unified Modeling Language®); CORBA® (Common Object Request Broker Architecture); CWM™ (Common Warehouse Metamodel™); and industry-specific standards for dozens of vertical markets.

More information on the OMG is available at https://www.omg.org/.

OMG Specifications

As noted, OMG specifications address middleware, modeling and vertical domain frameworks. All OMG Specifications are available from the OMG website at:

https://www.omg.org/spec

All of OMG's formal specifications may be downloaded without charge from our website. (Products implementing OMG specifications are available from individual suppliers.) Copies of specifications, available in PostScript and PDF format, may be obtained from the Specifications Catalog cited above or by contacting the Object Management Group, Inc. at:

OMG Headquarters
9C Medway Road, PMB 274
Milford, MA 01757
USA

Tel: +1-781-444-0404
Fax: +1-781-444-0320
Email: pubs@omg.org

Certain OMG specifications are also available as ISO standards. Please consult http://www.iso.org

Issues

The reader is encouraged to report any technical or editing issues/problems with this specification via the report form at:

https://issues.omg.org/issues/create-new-issue

1 Scope

1.1 Introduction

The Commons Ontology Library is designed to provide a useful set of modeling constructs that are reusable in different modeling and data deployment environments with minimal commitments. It is intended to be extensible such that new ontologies and potentially other models (for example, UML models corresponding to the ontologies) can be added as cross-domain requirements present themselves. These requirements may come from other OMG standards efforts or potentially from external users of the library, for example, the Industrial Ontology Foundry (IOF) manufacturing community, an EDM Council project with the Pistoia Alliance IDMP pharmaceutical community, and others.

1.2 Criteria for Inclusion

Ontologies and other models will be identified primarily by drawing on other work, although care must be taken to ensure that intellectual property and other legal rights are addressed and that standardization is desired by the user community. Oversight for curation of the library will be managed by the Commons task force (RTF) via the normal OMG process. The minimum criteria identified to date for inclusion include: (1) the need for the same set of concepts with the same semantics across multiple specifications and/or domain areas, such as manufacturing, finance and/or retail, (2) a clear set of use cases, competency questions, and test cases that can help limit the scope for a given ontology and provide the basis for regression testing, (4) reusability in their own right with minimal dependencies on other ontologies with the possible exception of other Commons ontologies, and (4) that the ontologies meet minimal requirements for metadata, logical consistency, and serialization (*e.g.*, RDF/XML and Turtle serialized OWL, for OWL ontologies).

1.3 Overview

The initial Commons Ontology Library of ontologies specified herein covers:

(1) Annotations

- a reusable set of declarations for commonly used annotation properties from the Dublin Core Metadata Initiative (DCMI) Terms[1] and the Simple Knowledge Organization System (SKOS)[2], so that these vocabularies can be reused without importing either, and
- additional annotation properties that provide metadata for documentation that is not explicitly available in either Dublin Core or SKOS.

(2) Collections:

- commonly used concepts for arrangements and schemes for organizing information and collections of things, such as structured collections that may be organized according to some scheme, and related very high level mereology relations to enable association of things with such collections and schemes.

(3) Designations:

[1] See https://www.dublincore.org/specifications/dublin-core/dcmi-terms/

[2] See https://www.w3.org/2004/02/skos/

- Designators – commonly used concepts for naming, derived in part from the patterns defined in ISO 1087 for terminology work and ISO 11179-3, Metadata Registries. The top-level designators ontology includes several very high level semiotic relationships, including defines, describes, and denotes for associating designators with the concepts they reference.

- Contextual Designators – an extension to the designators ontology to incorporate applicable dates and times and facilitate the inclusion of other context that is commonly needed, derived in part from the patterns defined in ISO 11179-3, Metadata Registries.

- Codes and Code Sets – commonly used concepts for describing codes, including standardized codes such as ISO language, country, and other code sets, the North American Industry Classification System (NAICS) codes, and custom code sets that many organizations develop for various purposes, derived from the patterns specified in ISO 11179-3, Metadata Registries.

- Identifiers – commonly used concepts for describing identifiers and the identification schemes that define them, such as various national and international identifiers for legal entities, financial instruments, and the like, derived from the patterns specified in ISO 11179-3, Metadata Registries.

- Contextual Identifiers – an extension to the contextual designators and identifiers ontologies covering concepts for describing more complex identifiers, including those that apply for some period of time as well as those that are structured and include other codes or identifiers.

(4) Classifiers:

- abstract concepts for representation of classification schemes that enable the classification of arbitrary concepts into hierarchies (or partial orders) for use in other ontologies, derived in part from the patterns defined in ISO 1087-1 for terminology work and ISO 11179-3, Metadata Registries.

(5) Time:

- Dates and Times – commonly used temporal concepts that cover those most frequently needed across domains, with a focus on terminology that is used in business applications. It is designed to be mappable to other date and time ontologies and specifications, such as the W3C Time Ontology in OWL[3], certain temporal elements in ISO Basic Formal Ontology[4], time concepts defined in schema.org, and the OMG Date Time Vocabulary (DTV) specification, without the corresponding overhead, or in some cases, issues. The concepts were originally derived from a number of date and time standards including ISO 8601:2004 Representation of Dates and Times.

- Mapping Dates and Times to OWL Time – an extension to the dates and times ontology to map it to the widely used W3C Time Ontology in OWL recommendation.

(6) Text Datatype:

- a custom datatype that combines language tagged and plain string values. This text datatype is useful in cases where it is not clear whether string values will be tagged or not, but where it is anticipated that multilingual strings might be appropriate.

Each of these ontologies are defined below.

---

3 Available at https://www.w3.org/TR/owl-time/

4 See https://basic-formal-ontology.org/bfo-2020.html

1.4 Metadata

Annotations on concepts, properties, and individuals in this specification follow the general policies recommended by the OMG Architecture Board, including the use of (1) the Dublin Core Metadata Terms [Dublin Core], (2) the Simple Knowledge Organization System (SKOS) [SKOS], and (3) the annotation vocabulary included in the Commons Ontology Library. Every element in the ontologies defined in the Commons Ontology Library must have a label and definition, and in many cases, the source for the definitions, such as an ISO or other OMG standard, is referenced. Examples are also included as appropriate, along with other notes that may assist users in understanding and reusing the ontology.

2 Conformance

The Commons Ontology Library specification provides two options for conformance points for implementers. These are as follows:

(1) Specification-level conformance with all of the RDF/OWL ontologies, which means that the subject application formally imports all of the ontologies (*i.e.*, through `owl:imports` statements in another ontology or via loading the full set of ontologies for reference in a knowledge base that supports RDF/OWL) with no resulting logical inconsistencies;

(2) Linked Data-level conformance – which means that the subject application references one or more of the ontologies but does not formally import them.

For either conformance point, references to the elements defined in a given ontology must use, or provide a mapping to, the standard OMG URI for that element. Users may choose to use or extend extend any of the Commons Ontology Library ontologies as necessary, to add concepts and properties required between releases, or to add application-specific extensions needed to address their individual requirements. We encourage library implementers and users to submit any requirements for extension, including requests to add ontologies to the library, to the relevant task force.

3 References

3.1 Normative References

Reference	***Description***
[BCP 47]	BCP 47: Tags for Identifying Languages, available at https://tools.ietf.org/search/bcp47
[DTV]	Date-Time Vocabulary (DTV™). Available at https://www.omg.org/spec/DTV/.
[Dublin Core]	DCMI Metadata Terms, Issued 2020-01-20 by the Dublin Core™ Metadata Initiative. Available at https://www.dublincore.org/specifications/dublin-core/dcmi-terms/.
[ISO 704]	ISO 704:2009 Terminology work – Principles and methods, Third edition, 2009-11-01
[ISO 1087]	ISO 1087:2019 Terminology work – Vocabulary – Theory and Application, Second edition, 2019-09
[ISO 8601-1]	ISO 8601-1:2019 Date and Time – Representations for information interchange – Part 1: Basic Rules
[ISO 11179-3]	ISO/IEC 11179-3:2013 Information technology – Metadata registries (MDR) – Registry metamodel and basic attributes, Third edition, 2013-02-15
[MOF]	Meta Object Facility (MOF™) Core. Available at http://www.omg.org/spec/MOF/
[MOF XMI]	MOF 2/XMI (XML Metadata Interchange) Mapping Specification. Available at http://www.omg.org/spec/XMI/
[ODM]	Ontology Definition Metamodel (ODM™). Available at http://www.omg.org/spec/ODM/
[OWL 2]	OWL 2 Web Ontology Language Quick Reference Guide (Second Edition), W3C Recommendation 11 December 2012. Available at http://www.w3.org/TR/2012/REC-owl2-quick-reference-20121211/.
[RDF Concepts]	RDF 1.1 Concepts and Abstract Syntax. Richard Cyganiak, David Wood and Markus Lanthaler, Editors. W3C Recommendation, 25 February 2014. Available at http://www.w3.org/TR/rdf11-concepts/
[RDF Schema]	RDF Schema 1.1. Dan Brickley and R.V. Guha, Editors. W3C Recommendation, 25 February 2014. Available at http:// www.w3.org/TR/rdf-schema/.
[SKOS]	SKOS Simple Knowledge Organization System Reference, W3C Recommendation 18 August 2009. Available at http://www.w3.org/TR/2009/REC-skos-reference-20090818/.
[SMOF]	MOF Support for Semantic Structures (SMOF™). Available at https://www.omg.org/spec/SMOF/.
[SysML]	OMG System Modeling Language (SysML®). Available at https://www.omg.org/spec/SysML/.
[UML]	Unified Modeling Language™ (UML®). Available at http://www.omg.org/spec/UML/
[Unicode]	*The Unicode Standard, Version 3*, The Unicode Consortium, Addison-Wesley, 2000. ISBN 0-201-61633-5, as updated from time to time by the publication of new

	versions. (See http:// www.unicode.org/unicode/standard/versions/ for the latest version and additional information on versions of the standard and of the Unicode Character Database).
[UTF-8]	RFC 3629: UTF-8, a transformation format of ISO 10646. F. Yergeau. IETF, November 2003, http://www.ietf.org/rfc/rfc3629.txt
[W3C Datatypes in RDF and OWL]	XML Schema Datatypes in RDF and OWL, W3C Working Group Note 14 March 2006, Available at http://www.w3.org/TR/2006/NOTE-swbp-xsch-datatypes-20060314/.
[W3C OWL Time]	W3C Time Ontology in OWL, available at https://www.w3.org/TR/owl-time/
[XML Schema Datatypes]	XML Schema Part 2: Datatypes Second Edition. W3C Recommendation 28 October 2004. Available at http://www.w3.org/TR/xmlschema-2/.

3.2 Non-Normative References

The following informative documents are referenced in this specification:

Reference	***Description***
[DL Handbook]	THE DESCRIPTION LOGIC HANDBOOK: Theory, implementation, and applications. Baader, McGuinness, Nardi, and Patel-Schneider, editors. Cambridge University Press, Cambridge, United Kingdom, 2003.
[OE]	Kendall, Elisa F. and Deborah L. McGuinness. *Ontology Engineering*: Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool Publishers. 2019. doi: 10.2200/S00834ED1V01Y201802WBE018

4 Terms and Definitions

For the purposes of this specification, the following terms and definitions apply.

Term	*Definition*
ontology	An ontology specifies a rich description of the • Terminology, concepts, nomenclature • Relationships among and between concepts and individuals • Sentences distinguishing concepts, refining definitions and relationships (constraints, restrictions, regular expressions) relevant to a particular domain or area of interest. [OE]

5 Symbols

5.1 Symbols

See clause 6.5, Notation, for a description of the logic symbols used to describe the ontologies covered in this specification.

5.2 Abbreviations

The following abbreviations are used throughout this specification:

DL – Description Logics

FIBO – Financial Industry Business Ontology

IOF – Industrial Ontology Foundry

IRI – Internationalized (Uniform) Resource Identifier

ISO – International Organization for Standardization

LCC – Languages, Countries and Codes

MVF – Multiple Vocabulary Facility

OWL – Web Ontology Language

ODM – Ontology Definition Metamodel

RDF – Resource Definition Framework

UML – Unified Modeling Language

URI – Uniform Resource Identifier

URL – Uniform Resource Locator

W3C – World Wide Web Consortium

XMI – XML Metadata Interchange

XML – eXtensible Markup Language

6 Additional Information

6.1 Changes to Other OMG Specifications

None.

6.2 Acknowledgments

The following organization submitted this specification:

- Thematix Partners LLC

The following companies and organizations are supporters of this specification:

- agnos.ai U.K. Ltd
- EDM Council, Inc.
- Mayo Clinic
- Pistoia Alliance, Inc.
- Raytheon Technologies
- Rensselaer Polytechnic Institute
- U. S. National Institute of Standards and Technology (NIST)
- Wells Fargo Bank, N.A.
- Working Ontologist

6.3 Intellectual Property Rights

The Commons Ontology Library is available under the OMG's Copyright and Non-Assertion Covenant (see https://www.omg.org/cgi-bin/doc.cgi?ipr for details). The individual ontologies are also licensed for use under the MIT open-source license agreement, available at http://opensource.org/licenses/MIT.

6.4 Application of the Commons Ontologies

The ontologies included in the library are reused by the Multiple Vocabulary Facility (MVF) specification and an anticipated update of the Languages, Countries and Codes (LCC) specification. With respect to LCC, they replace a number of existing concepts that were needed for MVF but derived from LCC. The ontologies are also needed for finalization of the API4KP specification. We anticipate that they will also be used in the next major revision to the Financial Industry Business Ontology (FIBO), in the emerging Retail Industry Ontology (RIO), and possibly others such as the Robotics Service Ontology specification.

In addition to their use in OMG standards, initiatives such as the Industrial Ontology Foundry (IOF), sponsored by the U.S. National Institute of Standards and Technology and a joint effort of the Pistoia Alliance and EDM Council for ontologies to facilitate identification of medicinal products (IDMP) are considering or currently using them as well.

6.5 Notation

The diagrams included herein are ODM-compliant UML diagrams. In other words, they conform to the UML Profiles for RDF and OWL specified in the OMG's Ontology Definition Metamodel [ODM] Specification. This includes the set of UML stereotypes and graphical notation used in the diagrams provided.

The color scheme employed in these diagrams includes:

- Basic OWL Classes: white for classes defined within the current (local) ontology, amber for classes defined within an imported (referenced) ontology
- OWL Restriction Classes and other Class Expressions (unions, intersection, complements): green
- OWL Object Properties: blue
- OWL Data Properties: dark gray
- OWL Datatypes: pink
- OWL Individuals: light gray

These colors are provided for clarification purposes only, and are non-normative.

For the library there is an "about" file, which provides metadata about the library, described below in tabular form. The ontologies themselves are documented as ODM-compliant UML models, aside from the "about" file, annotation vocabulary, and mapping ontology. Every ontology is expressed in RDF/XML-serialized OWL and Turtle-serialized OWL [OWL 2].

The notation used to represent description logic expressions (*i.e.*, the expressions in the Parent columns in class tables containing ontology details) is consistent with the notation defined in the Description Logic Handbook [DL Handbook]. The notation used in this specification, representing a subset of OWL 2, is described in Table 6.1, below.

Table 6.1: Description Logic Expressions Notation

Construct	*Description*	*Notation*
Boolean Connectives and Enumeration		
intersection	The intersection of two classes consists of exactly those individuals which are instances of both classes.	$C \cap D$
union	The union of two classes contains every individual which is contained in at least one of these classes.	$C \cup D$
enumeration	An enumeration defines a class by enumerating all its instances.	oneOf (i_1, i_2, i_3, … i_n)
Property Restrictions		
universal quantification	Universal quantification is used to specify a class of individuals for which all related individuals must be instances of a given class (*i.e.*, allValuesFrom in OWL).	$\forall$R.C, where R is the relation (property) and C is the class that constrains all values for related individuals
existential quantification	Existential quantification is used to specify a class as the set of all individuals that are connected via a particular	$\exists$R.C, where R is the relation (property) and C is the class that

	property to at least one individual which is an instance of a certain class (*i.e.*, someValuesFrom in OWL).	constrains some values of related individuals
individual value	Individual value restrictions are used to specify classes of individuals that are related to one particular individual (*i.e.*, hasValue in OWL).	∀R.I, where R is the relation (property) and I is the individual
exact cardinality	Cardinality (number) restrictions specify classes by restricting the cardinality on the sets of fillers for roles (relationships, or properties in OWL). Exact cardinality restrictions restrict the cardinality of possible fillers to exactly the number specified.	= n R (for unqualified restrictions) = n R.C (for qualified restrictions, i.e., including onClass or on DataRange)
maximum cardinality	Maximum cardinality restrictions restrict the cardinality of possible fillers to at most the number specified (inclusive).	≤ n R (for unqualified restrictions) ≤ n R.C (for qualified restrictions)
minimum cardinality	Minimum cardinality restrictions restrict the cardinality of possible fillers to at least the number specified (inclusive).	≥ n R (for unqualified restrictions) ≥ n R.C (for qualified restrictions)
Class Axioms		
equivalent classes	Two classes are considered equivalent if they contain exactly the same individuals.	≡ C
disjoint classes	Disjointness means that membership in one class specifically excludes membership in another.	¬ C
Property Axioms		
complex role inclusions	Role inclusions allow [object] properties to be chained together in a sequence that is a subproperty of a higher-level property.	R ○ R

Note that in the case of complex restrictions, where there are nested elements in parentheses, the “dot notation” used as a separator between a property and the role filler is replaced with the embedded parenthetical filler definition. A “role” from a description logic perspective is essentially a property in OWL, and the role “filler” is the class or individual that provides the value for that role in a given axiom (*i.e.*, in a restriction or other logic expression).

7 Architecture

7.1 "About" the Commons Ontologies

The "about" file for the Commons Ontology Library provides metadata describing the library. This file is designed to (1) describe the machine-readable content of the specification for users that download the entire library directly and imports it into tools that can interpret and display the files, (2) for potential use in tagging the specification document on the OMG site, and (3) to provide a single file that imports the ontologies for ease of use (similar to a "make file" for software), excluding the mapping to the W3C Time Ontology in OWL, which may or may not be desired.

7.2 Namespace Definitions

The namespaces and prefixes corresponding to external elements required for use in the Commons Ontology Library are provided in Table 7.1. Table 7.2 provides the namespace declarations required for use of the ontologies included in the library itself. The prefixes provided in Tables 7.1 and 7.2 are normative, and their use is required in any conformant application or extension.

Table 7.1: Prefix and Namespaces for referenced/external vocabularies

Namespace Prefix	Namespace
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	http://www.w3.org/2000/01/rdf-schema#
owl	http://www.w3.org/2002/07/owl#
xsd	http://www.w3.org/2001/XMLSchema#
dct	http://purl.org/dc/terms/
skos	http://www.w3.org/2004/02/skos/core#
time	http://www.w3.org/2006/time#

The namespace approach taken for Commons Ontology Library is based on OMG guidelines and is constructed as follows:

- The standard protocol, authority, and top level specification part of any OMG specification namespace, which is `https://www.omg.org/spec/`
- The abbreviation for the specification: in this case `Commons`
- The ontology name

Note that the URI/IRI strategy for the ontologies included in the library takes a "slash" rather than "hash" approach, in order to accommodate server-side applications. Namespace prefixes are constructed as follows with the components separated by "-":

- The abbreviation used for prefix purposes across the Commons Ontology Library: `cmns`
- An abbreviation for the ontology name

The namespaces and prefixes for the individual ontologies are summarized in Table 7.2. These are given in alphabetical order, rather than with any intent to show imports relationships. The table includes the namespace definitions for the "about" file that is part of the machine-readable deliverables for the specification, but that is not required for imports closure. Note that these are not versioned, although version IRIs are included in every OWL ontology and are documented in the metadata for each of them.

Table 7.2: Prefix and Namespaces for the Commons Ontology Library Ontologies

Namespace Prefix	Namespace
abt-cmns	https://www.omg.org/spec/Commons/AboutCommons/
cmns-av	https://www.omg.org/spec/Commons/AnnotationVocabulary/
cmns-cds	https://www.omg.org/spec/Commons/CodesAndCodeSets/
cmns-cls	https://www.omg.org/spec/Commons/Classifiers/
cmns-col	https://www.omg.org/spec/Commons/Collections/
cmns-cxtdsg	https://www.omg.org/spec/Commons/ContextualDesignators/
cmns-cxtid	https://www.omg.org/spec/Commons/ContextualIdentifiers/
cmns-dsg	https://www.omg.org/spec/Commons/Designators/
cmns-dt	https://www.omg.org/spec/Commons/DatesAndTimes/
cmns-id	https://www.omg.org/spec/Commons/Identifiers/
cmns-mdt	https://www.omg.org/spec/Commons/MappingDatesAndTimesToOWLTime/
cmns-txt	https://www.omg.org/spec/Commons/TextDatatype/

8 Commons Ontologies

8.1 Ontology: Annotation Vocabulary

The annotation vocabulary provides commonly used annotation properties for documentation to facilitate understanding. It declares a number of properties available in the Dublin Core Metadata Initiative (DCMI)'s Metadata Terms vocabulary [Dublin Core] as OWL annotation properties to facilitate their usage in tools that require such declarations. It also declares the annotations provided in the Simple Knowledge Organization System [SKOS] to enable reuse without requiring import of the SKOS vocabulary, which includes semantics that may not be desirable for some knowledge graph applications. Finally, the vocabulary defines additional annotation properties that are useful for documenting other ontologies and are used in a number of OMG specifications.

Given that this ontology contains no classes, we have opted not to present a UML diagram for it herein. The metadata for this ontology is provided in Table 8.1, below and definitions for the new annotation properties (*i.e.*, those that are local to this ontology rather than declarations for Dublin Core and SKOS annotations) are presented in Table 8.2.

Table 8.1: Annotation Vocabulary Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/AnnotationVocabulary/
rdfs:label	Annotation Vocabulary
dct:abstract	The Annotation Vocabulary provides commonly used annotation properties for documentation to facilitate understanding.
cmns-av:copyright	Copyright (c) 2022 EDM Council, Inc.
cmns-av:copyright	Copyright (c) 2022 Object Management Group, Inc.
dct:license	http://opensource.org/licenses/MIT
dct:references	http://purl.org/dc/terms/
dct:references	http://www.w3.org/2004/02/skos/core#
dct:title	Commons Annotation Vocabulary
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/AnnotationVocabulary/
skos:note	Note that any of the annotation properties provided in Dublin Core can be used in addition to those declared herein. However, Dublin Core terms that are not explicitly defined herein must be declared explicitly as annotation properties in the ontologies that use them.
skos:note	The annotation properties defined below are derived from similar annotation vocabularies used in (1) the Object Management Group (OMG) specification metadata - see http://www.omg.org/techprocess/ab/SpecificationMetadata/, (2) annotations used in the Financial Industry Business Ontology (FIBO) - see

	`https://spec.edmcouncil.org/fibo/ontology/FND/Utilities/AnnotationVocabulary/, and (3) other ontology efforts such as the NIST-sponsored Industrial Ontology Foundation (IOF).`

Table 8.2: Annotation Vocabulary Details

Properties

Name	Annotations	Property Axioms
abbreviation (abbreviation)	Definition: designation formed by omitting parts from the full form of a term that denotes the same concept Note: Abbreviations can be created by removing individual words, or can be acronyms, initialisms, or clipped terms. Adapted from: ISO 1087 Terminology work and terminology science - Vocabulary, Second edition, 2019-09 Adapted from: ISO 31-0 Quantities and units - General principles Example: Chemical Symbols: H, O, Mg; Units of Measure: Km, Kg, G Explanatory note: The symbols for quantities are generally single letters of the Latin or Greek alphabet, sometimes with subscripts or other modifying signs. These letters, including those that are members of the Greek alphabet are not symbols for the purposes of this ontology, however, they are abbreviations. Expressions of chemical formulae may, however, include a combination of abbreviations and symbols, as needed to define a given quantity.	Parent Property: cmns-av:synonym
acronym (acronym)	Definition: abbreviation that is made up of the initial letters of the components of the full form of a term or proper name or from syllables of the full form Note: Acronyms are frequently pronounced syllabically. Adapted from: ISO 1087 Terminology work and terminology science - Vocabulary, Second edition, 2019-09 Example: Examples of acronyms are: laser, ISO, GATT, UNESCO, UNICEF	Parent Property: cmns-av:abbreviation
adaptedFrom (adapted from)	Definition: document or other source from which a given term (or its definition) was adapted (*i.e.*, is compatible with but not quoted); the range for this annotation can be a string, URI, or citation Usage note: This annotation should be used to indicate that a reference was used, for example, as input to the development of a definition or term but would not be considered infringing on a copyright.	Parent Property: dct:source
copyright (copyright)	Definition: exclusive legal right, given to an originator or an assignee to print, publish, perform, film, or record literary, artistic, or musical material, and to authorize others to do the same	Parent Property: dct:rights

	Usage note: This annotation is typically used to describe an artifact such as a controlled vocabulary, ontology, or other similar resource.	
directSource (direct source)	Definition: quoted reference for the subject resource; the range for this annotation can be a string, URI, or bibliographic citation	Parent Property: dct:source
explanatoryNote (explanatory note)	Definition: note that provides additional explanatory material for a resource	Parent Property: skos:note
logicalDefinition (logical definition)	Definition: definition in the form of a formal expression, such as the mathematical or logic representation, for the resource	Parent Property: skos:definition
symbol (symbol)	Definition: abbreviation that is a design or mark, or other non-alpha-numeric character(s) conventionally used to represent something, such as a currency or mathematical sign or operator	Parent Property: cmns-av:abbreviation
synonym (synonym)	Definition: designation that can be substituted for the primary representation of something Adapted from: ISO 1087 Terminology work and terminology science - Vocabulary, Second edition, 2019-09	Parent Property: skos:altLabel
usageNote (usage note)	Definition: note that provides information about how a given resource is used or may be extended	Parent Property: skos:note

8.2 Ontology: Classifiers

This ontology defines abstract concepts for representation of classification schemes that enable the classification of arbitrary concepts into hierarchies (or partial orders) for use in many other ontologies. It is derived in part from patterns defined in ISO 1087 for terminology work and ISO 11179-3, Metadata Registries.

Metadata for the Classifiers ontology is given in Table 8.3.

Table 8.3: Classifiers Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/Classifiers/
rdfs:label	Commons Classifiers Ontology
dct:abstract	This ontology defines abstract concepts for representation of classification schemes that enable the classification of arbitrary concepts into hierarchies (or partial orders) for use in many other ontologies, derived in part from the patterns defined in ISO 1087-1 for terminology work and ISO 11179-3, Metadata Registries.
cmns-av:copyright	Copyright (c) 2014-2022 EDM Council, Inc.
cmns-av:copyright	Copyright (c) 2014-2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2022 Object Management Group, Inc.

dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/Classifiers/
skos:note	The classifiers ontology conforms with the OWL 2 DL semantics, and is outside of OWL 2 RL due to the inclusion of a local some values constraint. The latter could be removed as needed to support OWL RL rule-based applications that cannot be extended to support it.
skos:note	This ontology was originally designed for use in the OMG Languages, Countries and Codes (LCC) specification as part of the broader CountryRepresentation ontology. The concepts have also been used in the Financial Industry Business Ontology (FIBO) for representing industry sectors, financial instrument classifiers (e.g., asset classes), lifecycle states, and so forth.

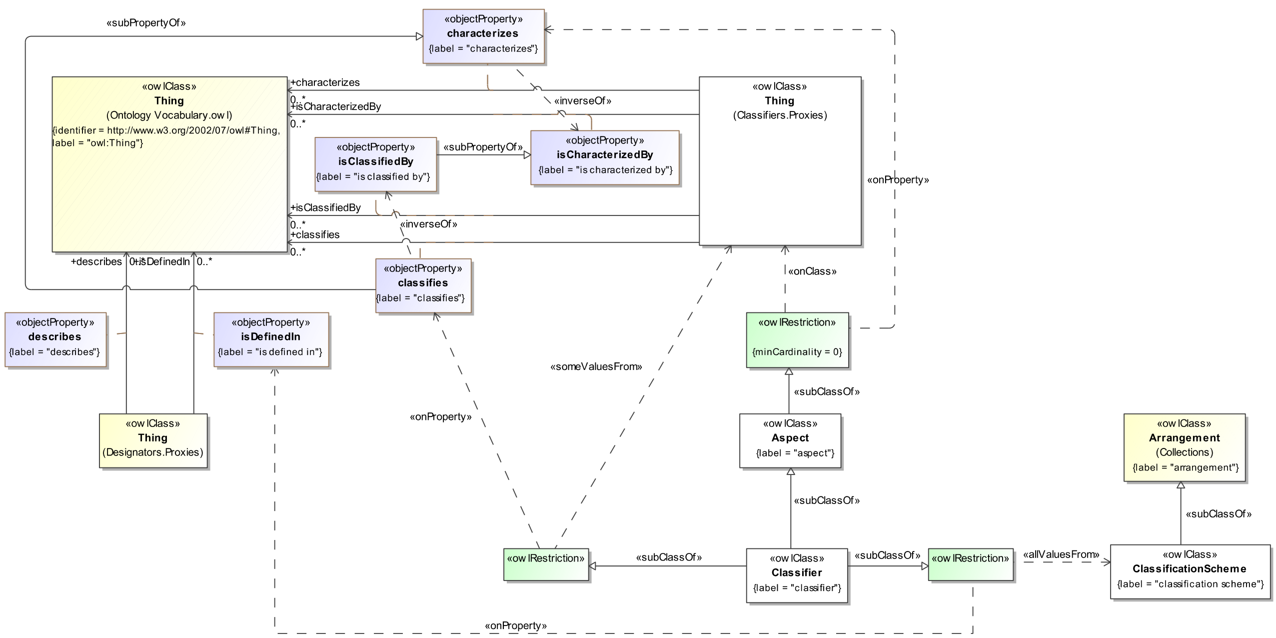

Figure 1: Overview of the Classifiers Ontology

An overview of the Classifiers ontology is given in Figure 1, above. The detailed annotations and axioms that comprise the Classifiers ontology are provided in Table 8.4, below.

Table 8.4:Classifiers Ontology Details

Classes

Name	Annotations	Class Expressions
Aspect (aspect)	Definition: characteristic or feature that can be used to dimensionalize, filter, or subset something Synonym: characteristic	Property Restriction: ≥ 0 characterizes
ClassificationScheme	Definition: system for allocating classifiers to	Parent Class: cmns-col:Arrangement

(classification scheme)	things Note: ISO 11179-3 defines a classification scheme as descriptive information for an arrangement or division of objects into groups based on criteria such as characteristics, which the objects have in common. A classification scheme may be a taxonomy, a network, an ontology, or any other terminological system. Such classification schemes are intended to permit the classification of arbitrary objects into hierarchies, or partial orders, as appropriate. The classification may also be just a list of controlled vocabulary of property words (or terms). The list might be taken from the 'leaf level' of a taxonomy. Source: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15 See also: https://en.wikipedia.org/wiki/UTF-8	
Classifier (classifier)	Definition: standardized classification or delineation for something, per some scheme for such delineation, within a specified context Note: In ISO 1087, classifiers form categories of characteristics that serve as the criterion of subdivision when establishing concept systems. Example: The classifier 'color' embraces characteristics being red, blue, green, etc. The classifier 'material' embraces characteristics made of wood, metal, etc. Source: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	Parent Class: Aspect Property Restriction: ∀ isDefinedIn.ClassificationScheme Property Restriction: ∃ classifies.Thing

Properties

Name	Annotations	Property Axioms
characterizes (characterizes)	Definition: provides a discriminating feature or quality of	Parent Property: cmns-dsg:describes
classifies (classifies)	Definition: arranges in categories according to shared characteristics	Parent Property: cmns-cls:characterizes
isCharacterizedBy (is characterized by)	Definition: indicates a quality or feature of something, distinguishing it from something else	Parent Property: cmns-dsg:isDescribedBy Inverse: characterizes
isClassifiedBy (is classified by)	Definition: is systematically grouped based on characteristics by	Parent Property: cmns-cls:characterizes Inverse: classifies

8.3 Ontology: Codes and Code Sets

The Codes and Code Sets ontology defines commonly used concepts for describing codes, including standardized codes such as ISO language, country, and other code sets, the North American Industry Classification System (NAICS) codes, and custom code sets that many organizations develop for various purposes, derived from the patterns specified in ISO 11179-3, Metadata Registries.

Metadata for the Codes and Code Sets ontology is given in Table 8.5.

Table 8.5: Codes and Code Sets Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/CodesAndCodeSets/
rdfs:label	Commons Codes and Code Sets Ontology
dct:abstract	This ontology defines commonly used concepts for describing codes, including standardized codes such as ISO language, country, and other code sets, the North American Industry Classification System (NAICS) codes, and custom code sets that many organizations develop for various purposes, derived from the patterns specified in ISO 11179-3, Metadata Registries.
dct:contributor	Elisa Kendall, Thematix Partners LLC
dct:contributor	Pete Rivett, agnos.ai
cmns-av:copyright	Copyright (c) 2014-2022 EDM Council, Inc.
cmns-av:copyright	Copyright (c) 2014-2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2021-2022 agnos.ai U.K. Ltd
cmns-av:copyright	Copyright (c) 2022 Object Management Group, Inc.
dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/CodesAndCodeSets/
skos:note	The codes and code sets ontology conforms with the OWL 2 DL semantics, and is outside of OWL 2 RL due to (1) imported axioms from the designations ontology, and (2) the inclusion of a local some values constraint. The latter could be removed as needed to support OWL RL rule-based applications that cannot be extended to support it.
skos:note	This ontology was originally designed for use in the OMG Languages, Countries and Codes (LCC) specification as part of the broader LanguageRepresentation ontology. The concepts have also been used in the Financial Industry Business Ontology (FIBO) for representing currency codes, market identifiers (MIC codes), codes for corporate

	`actions, and so forth.`

An overview of the Codes and Code Sets ontology is given in Figure 2.

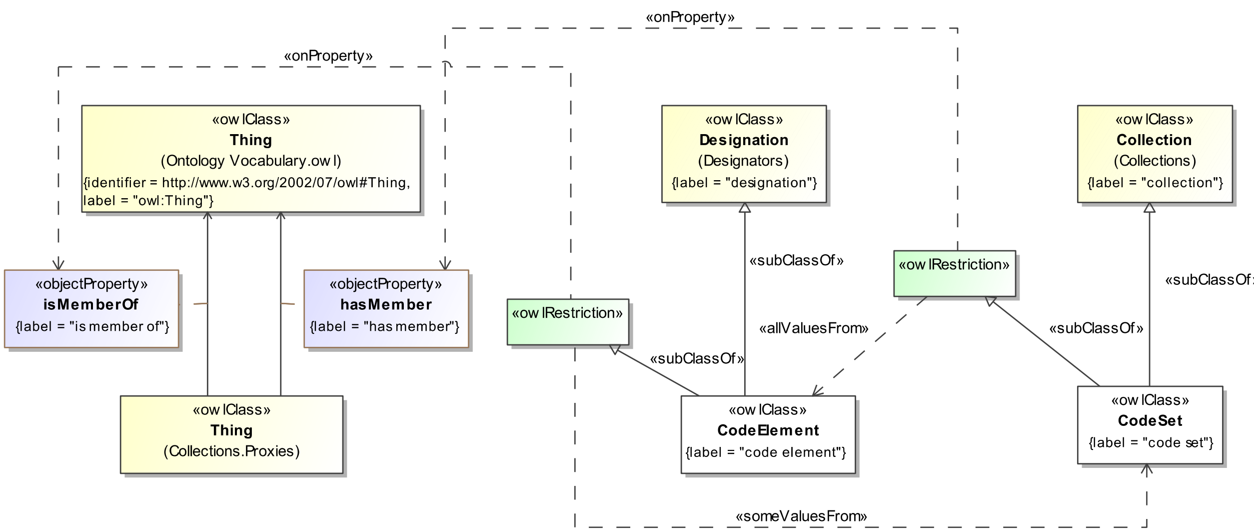

Figure 2: Overview of the Codes and Code Sets Ontology

The detailed annotations and axioms that comprise the Codes and Code Sets ontology are provided in Table 8.6, below.

Table 8.6: Codes and Code Sets Ontology Details

Classes

Name	Annotations	Class Expressions
CodeElement (code element)	Definition: sequence of characters denoting something for some purpose, within a specified context, according to some rule set Note: Note that codes may be included in multiple code lists, especially in cases where there are multiple versions of those code lists. ICD-9 and ICD-10 are examples of code sets that specify, in some cases, the same codes, but across different versions of those code sets. Example: An example of a code set that has multiple versions are the International Statistical Classification of Diseases and Related Health Problems (ICD) codes such as ICD-9, ICD-10, and so forth, that specify the same codes across multiple versions. Source: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	Parent Class: cmns-dsg:Designation Property Restriction: ∃ cmns-col:isMemberOf.CodeSet
CodeSet (code set)	Definition: system of alpha-numeric symbols, or combinations of symbols, that stand for specified	Parent Class: cmns-col:Collection

	values in some context Synonym: code system Source: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	Property Restriction: ∀ cmns-col:hasMember

8.4 Ontology: Collections

The collections ontology defines commonly used concepts for arrangements and schemes for organizing information and collections of things, such as structured collections that may be organized according to some scheme, and related very high level mereology relations to enable association of things with such collections and schemes.

Metadata for the Collections ontology is given in Table 8.7.

Table 8.7: Collections Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/Collections/
rdfs:label	Commons Collections Ontology
dct:abstract	The collections ontology defines commonly used concepts for arrangements and schemes for organizing information and collections of things, such as structured collections that may be organized according to some scheme, and related very high level mereology relations to enable association of things with such collections and schemes.
dct:contributor	Davide Sottara, Mayo Clinic
dct:contributor	Elisa Kendall, Thematix Partners LLC
dct:contributor	Pete Rivett, agnos.ai
cmns-av:copyright	Copyright (c) 2019-2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2021-2022 agnos.ai U.K. Ltd
cmns-av:copyright	Copyright (c) 2021-2022 EDM Council, Inc.
cmns-av:copyright	Copyright (c) 2021-2022 Mayo Clinic
cmns-av:copyright	Copyright (c) 2022 Object Management Group, Inc.
dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/Collections/
skos:note	The collections ontology conforms with the OWL 2 DL semantics, and is outside of OWL 2 RL due to the inclusion of a local some values constraint. This restriction may be removed as needed to support OWL RL rule-based

	applications that cannot be extended to support it.
skos:note	This ontology was originally designed for use in the OMG Languages, Countries and Codes (LCC) specification as part of the broader LanguageRepresentation ontology. The concepts have also been used in the Financial Industry Business Ontology (FIBO) for representing collections such as baskets, portfolios records, statistical universes and populations, etc., and schemes such as classification schemes and identification schemes.

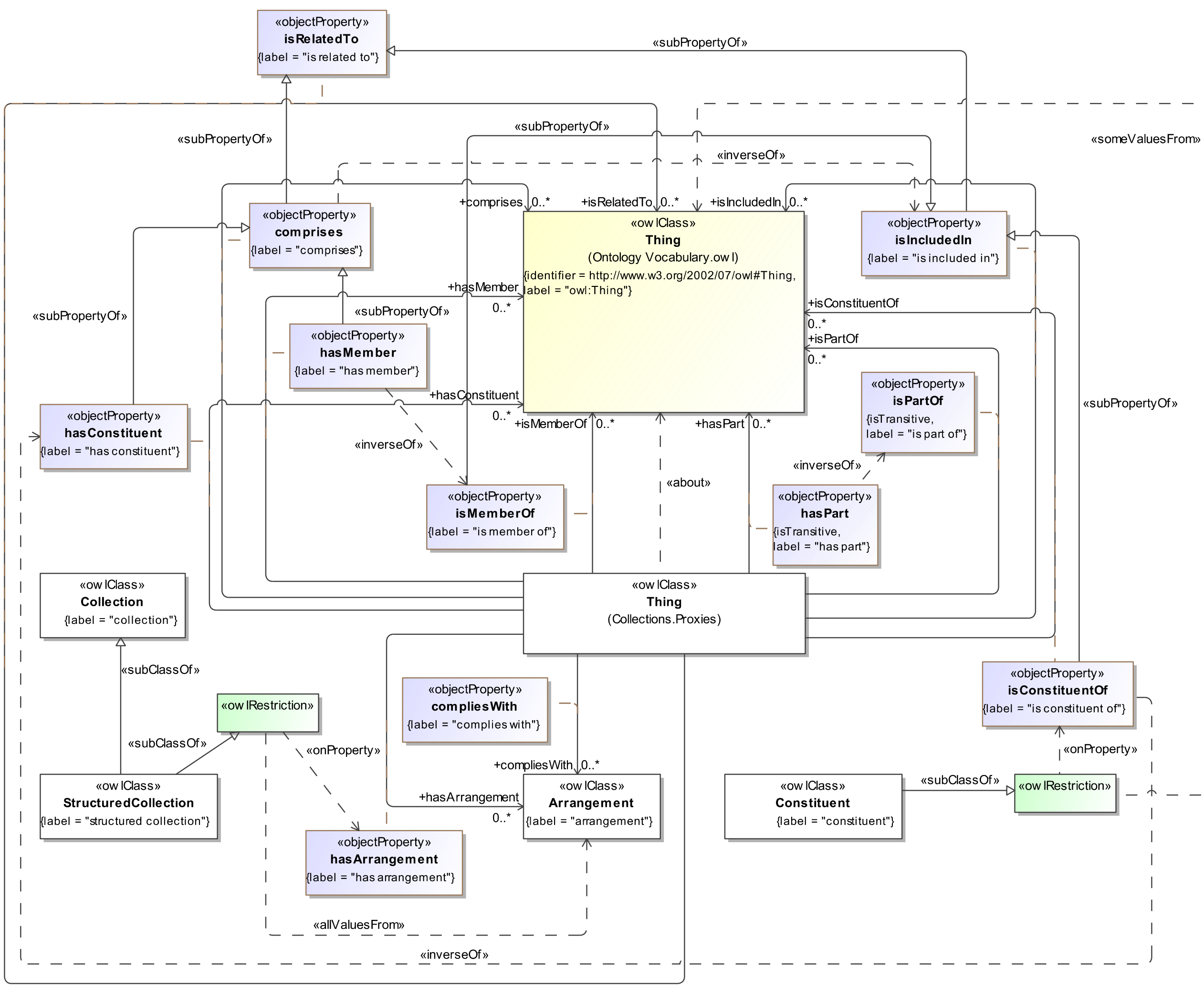

Figure 3: Overview of the Collections Ontology

An overview of the Collections ontology is given in Figure 3.

The detailed annotations and axioms that comprise the Collections ontology are provided in Table 8.8, below.

Table 8.8: Collections Ontology Details

Classes

Name	Annotations	Class Expressions
Arrangement (arrangement)	Definition: systematic plan, manner, or method for making, doing, achieving, or organizing something Example: Examples include designs, schema, models, methodologies, alphabetical or numeric ordering, and the like.	
Collection (collection)	Definition: grouping of things (may be zero) that have some shared significance	
Constituent (constituent)	Definition: component of a substance, collection or combination of things	Property Restriction: ∃ isConstituentOf.owl:Thing
StructuredCollection (structured collection)	Definition: collection that has a clearly defined structure or organization Example: Examples include collections organized thematically, alphabetically, by method used do develop them, according to time and/or version, or based on encoding schemes such as the Dewey Decimal System or Library of Congress Subject Headings.	Parent Class: cmns-col:Collection Property Restriction: ∀ hasArrangement.Arrangement

Properties

Name	Annotations	Property Axioms
compliesWith (complies with)	Definition: adheres to policies or rules specified in	Range: Arrangement
comprises (comprises)	Definition: includes, consists of, or contains, especially within a particular scope Note: Note that something can be comprised of something(s) that may or may not be understood as separable parts. In other words, comprises does not imply countability or uniqueness.	Parent Property: isRelatedTo
hasArrangement (has arrangement)	Definition: is structured or organized according to	Range: Arrangement
hasConstituent (has constituent)	Definition: consists of or contains Usage note: Being a constituent of something does not necessarily mean parthood. Whole-part relations are transitive, whereas constituency is not necessarily transitive and so this property is useful in cases where transitivity is not necessarily desirable or appropriate.	Parent Property: comprises
hasMember (has member)	Definition: includes, as a discrete element	Parent Property: comprises

	Note: Note that the domain of hasMember should be some sort of collection, aggregate, or group. In the Financial Industry Business Ontology (FIBO), hasMember is used in the case of parties (people and organizations), whereas comprises can have anything in its range.	
hasPart (has part)	Definition: indicates any portion of something, regardless of whether the portion itself is attached to the remainder or detached; cognitively salient or arbitrarily demarcated; self-connected or disconnected; homogeneous or gerrymandered; material or immaterial; extended or unextended; spatial or temporal Note: Note that 'has part' is not a subproperty of 'comprises' in order to enable transitivity for whole-part relationships without limiting the use of cardinality constraints on comprises and membership. Source: Stanford Encyclopedia of Philosophy at http://plato.stanford.edu/entries/mereology/	Type: owl:TransitiveProperty
isConstituentOf (is constituent of)	Definition: is a component of something else Note: A constituent may be an independently identifiable, discrete element or may be an indistinguishable element once it is combined with the target, such as a part of a substance.	Parent Property: isIncludedIn Inverse: hasConstituent
isIncludedIn (is included in)	Definition: is contained in or an element of	Parent Property: isRelatedTo Inverse: comprises
isMemberOf (is member of)	Definition: is a discrete element of	Parent Property: isIncludedIn Inverse: hasMember
isPartOf (is part of)	Definition: relates something to another thing that it is some component or portion of, regardless of how that whole-part relationship is manifested Note: Note that 'is part of' is not a subproperty of 'is included in' in order to enable transitivity for whole-part relationships without limiting the use of cardinality constraints on inclusion and membership. Source: Stanford Encyclopedia of Philosophy at http://plato.stanford.edu/entries/mereology/	Type: owl:TransitiveProperty Inverse: hasPart
isRelatedTo (is related to)	Definition: links something or someone to something or someone else	

8.5 Ontology: Contextual Designators

The contextual designators ontology extends the designators ontology to incorporate applicable dates and times and facilitate the inclusion of other context that is commonly needed, derived in part from the patterns defined in ISO 11179-3, Metadata Registries.

Metadata for the Contextual Designators ontology is given in Table 8.9.

Table 8.9: Contextual Designators Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/ContextualDesignators/
rdfs:label	Commons Contextual Designators Ontology
dct:abstract	The contextual designators ontology extends the designators ontology to incorporate applicable dates and times and facilitate the inclusion of other context that is commonly needed, derived in part from the patterns defined in ISO 11179-3, Metadata Registries.
dct:contributor	Dean Allemang, Working Ontologist
dct:contributor	Elisa Kendall, Thematix Partners LLC
dct:contributor	Pete Rivett, agnos.ai
cmns-av:copyright	Copyright (c) 2020-2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2020-2022 Working Ontologist LLC
cmns-av:copyright	Copyright (c) 2022 agnos.ai U.K. Ltd
cmns-av:copyright	Copyright (c) 2022 Object Management Group, Inc.
dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/ContextualDesignators/
skos:note	The contextual designators ontology conforms with the OWL 2 DL semantics, and is outside of OWL 2 RL due to (1) imported axioms from the designations and dates and times ontologies, and (2) the inclusion of local some values and min 0 cardinality constraints. The latter could be removed as needed to support OWL RL rule-based applications that cannot be extended to support it.

An overview of the Contextual Designators ontology is given in Figure 4.

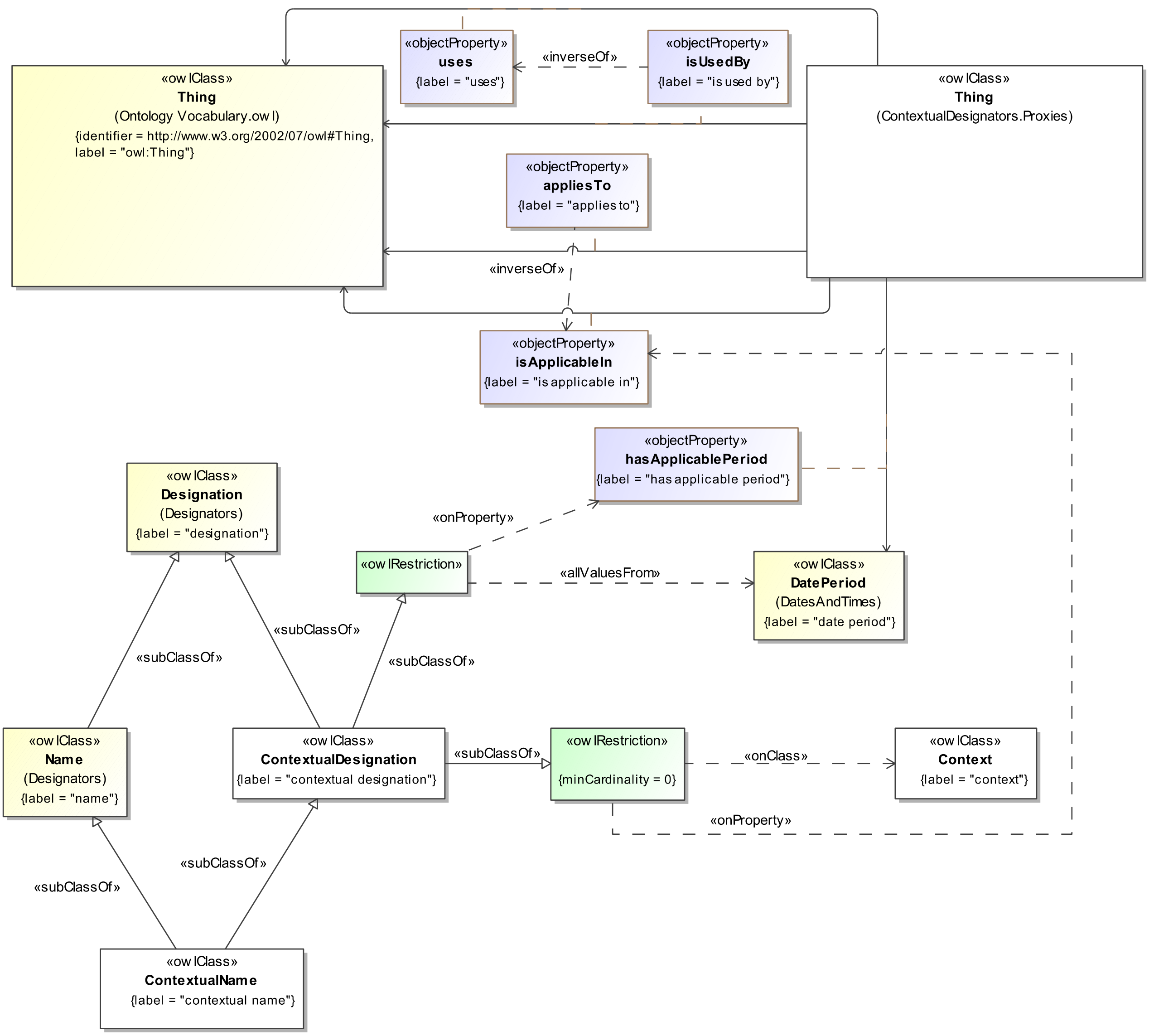

Figure 4: Overview of the Contextual Designators Ontology

The detailed annotations and axioms that comprise the Contextual Designators ontology are provided in Table 8.10, below.

Table 8.10: Contextual Designators Ontology Details

Classes

Name	Annotations	Class Expressions
Context (context)	Definition: situation or frame of reference in	

	which something applies, exists, happens, or is used and that helps to illustrate or explain it Note: From a terminology perspective, context provides information, including but not limited to text, that illustrates a concept or the use of a designation for a given situation. Source: ISO 1087 Terminology work and terminology science - Vocabulary, Second edition, 2019-09, clause 3.6.5 Source: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	
ContextualDesignation (contextual designation)	Definition: designation that applies to something in some context Note: Contextual designators may be structured such that they include other designators, for example, composite identifiers that include a country code to distinguish national identifiers from one another, for example, in the case of some manufacturing, agricultural, or financial instrument identifiers. Note: Note that the use of the min 0 cardinality restriction in the definition of this class is provided as a reminder that contextual designators are expected, in most cases, to have some sort of context associated with them. There may be cases where the context is limited to a time period, though, and thus additional context may not be required, or where more direct relationships to provenance, governance, or other contextual information is available.	Parent Class: cmns-dsg:Designation Property Restriction: ∀ hasApplicablePeriod.cmns-dt:DatePeriod Property Restriction: ≥ 0 isApplicableIn.Context
ContextualName (contextual name)	Definition: designation by which someone, some place, or something is known in some context Note: Names for people may be considered to be personally identifying information (PII), especially when other details are also available. Specifying names as string values attached directly to an individual makes name reconciliation and management, including from a privacy perspective, more challenging. Note: Names of people, places, and organizations often change over time, and may be used in a particular context, such as a DBA name for a business or legal name for a person. Note: This class is designed to be extended to include provenance details regarding the source for a particular name as well as links to the various contexts in which it is used.	Parent Class: ContextualDesignation, cmns-dsg:Name

Properties

Name	Annotations	Property Axioms
appliesTo (applies to)	Definition: indicates something for which a context is material, germane, or relevant in some way	
hasApplicablePeriod (has applicable period)	Definition: indicates a date period during which something may be used, applies, is valid or is accurate or relevant	Parent Property: isApplicableIn, cmns-dt:hasDatePeriod Range: cmns-dt:DatePeriod
isApplicableIn (is applicable in)	Definition: indicates a context in which something is relevant	Inverse: appliesTo
isUsedBy (is used by)	Definition: is employed in the process of accomplishing something for	Inverse: uses
uses (uses)	Definition: employs as a means of accomplishing some task or achieving some result	

8.6 Ontology: Contextual Identifiers

The contextual identifiers ontology defines commonly used concepts for describing more complex identifiers, including those that apply for some period of time as well as those that are structured and include other codes or identifiers.

Metadata for the Contextual Identifiers ontology is given in Table 8.11.

Table 8.11: Contextual Identifiers Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/ContextualIdentifiers/
rdfs:label	Commons Contextual Identifiers Ontology
dct:abstract	The contextual identifiers ontology defines commonly used concepts for describing more complex identifiers, including those that apply for some period of time as well as those that are structured and include other codes or identifiers.
dct:contributor	Elisa Kendall, Thematix Partners LLC
dct:contributor	Evan Wallace, U.S. National Institute of Standards and Technology (NIST)
cmns-av:copyright	Copyright (c) 2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2022 Object Management Group, Inc.
dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/

	ContextualIdentifiers/

An overview of the Contextual Identifiers ontology is given in Figure 5.

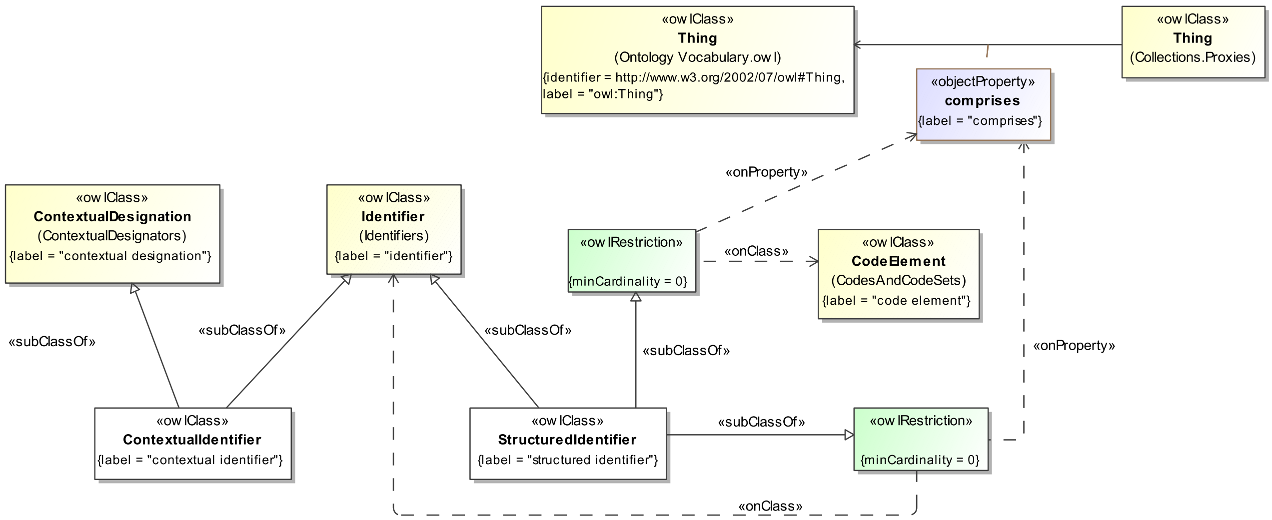

Figure 5: Overview of the Contextual Identifiers Ontology

The detailed annotations and axioms that comprise the Contextual Identifiers ontology are provided in Table 8.12, below.

Table 8.12: Contextual Identifiers Ontology Details

Classes

Name	Annotations	Class Expressions
ContextualIdentifier (contextual identifier)	Definition: sequence of characters uniquely identifying that with which it is associated, within a specified context Note: The context within which an identifier is unique may be limited to a given data source, registry or jurisdiction, or may be designed to be globally unique such as a legal entity identifier issued by a registrar authorized by the Global LEI Foundation. Such identifiers may have other features associated with them, such as the date they were originally issued, and information related to registration, validation, recency, and so forth.	Parent Class: cmns-cxtdsg:ContextualDesignation, cmns-id:Identifier
StructuredIdentifier (structured identifier)	Definition: sequence of characters uniquely identifying that with which it is associated, that includes other codes or identifiers, or that is constructed from other notions Note: Many structured identifiers can be validated using a regular expression, such as a social security number in the United States.	Parent Class: cmns-id:Identifier Property Restriction: ≥ 0 cmns-col:comprises.cmns-cds:CodeElement Property Restriction: ≥ 0 cmns-col:comprises.cmns-id:Identifier

	Example: A vehicle identification number (VIN) includes a world-wide manufacturer identifier, a vehicle description (*i.e.*, make, model), check digits, the year, plant and a specific vehicle number. Example: An international security identification number (ISIN) includes a country code and the national security identification number (NSIN), as defined in ISO 6166.	

8.7 Ontology: Dates and Times

The dates and times ontology defines commonly used temporal concepts that cover those most frequently needed across domains, with a focus on terminology that is used in business applications. It is designed to be mappable to other date and time ontologies and specifications, such as the W3C Time Ontology in OWL[5], certain temporal elements in the Basic Formal Ontology (BFO 2020)[6], time concepts defined in schema.org, and the OMG's Date Time Vocabulary (DTV) specification[7], without the corresponding overhead or in some cases, issues. The concepts were originally derived from a number of date and time standards including ISO 8601:2004 Representation of Dates and Times. The ontology itself was derived from the Financial Industry Business Ontology (FIBO) Financial Dates ontology, with minor revisions to better reflect requirements for mapping to other ontologies.

Metadata for the Dates and Times ontology is given in Table 8.13.

Table 8.13: Dates and Times Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/DatesAndTimes/
rdfs:label	Commons Dates and Times Ontology
dct:abstract	The dates and times ontology defines commonly used temporal concepts that cover those most frequently needed across domains, with a focus on terminology that is used in business applications. It is designed to be mappable to other date and time ontologies and specifications, such as the W3C Time Ontology in OWL (available at https://www.w3.org/TR/owl-time/), certain temporal elements in BFO 2020 (see https://basic-formal-ontology.org/bfo-2020.html), time concepts defined in schema.org, and the Object Management Group's Date Time Vocabulary (DTV) specification (available at https://www.omg.org/spec/DTV/), without the corresponding overhead or in some cases, issues. The concepts were originally derived from a number of date and time standards including ISO 8601:2004 Representation of Dates and Times. The ontology itself was derived from the Financial Industry Business Ontology (FIBO) Financial Dates ontology, with minor revisions to better reflect requirements for mapping to other ontologies.

5 See https://www.w3.org/TR/owl-time/

6 See https://basic-formal-ontology.org/bfo-2020.html

7 Available at https://www.omg.org/spec/DTV/

dct:contributor	Elisa Kendall, Thematix Partners LLC
dct:contributor	Mark Linehan, Thematix Partners LLC
dct:contributor	Pete Rivett, agnos.ai
cmns-av:copyright	Copyright (c) 2014-2022 EDM Council, Inc.
cmns-av:copyright	Copyright (c) 2014-2022 Object Management Group, Inc.
cmns-av:copyright	Copyright (c) 2014-2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2021-2022 agnos.ai U.K. Ltd
dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/DatesAndTimes/
skos:note	The dates and times ontology conforms with the OWL 2 DL semantics, and is outside of OWL 2 RL due to the inclusion of exact cardinality constraints on explicit date, explicit duration and time of day. These constraints can be changed to maximum cardinality constraints if needed to support OWL RL rule-based applications that cannot be extended to support them.

The class hierarchy for the Dates and Times ontology is shown in Figure 6.

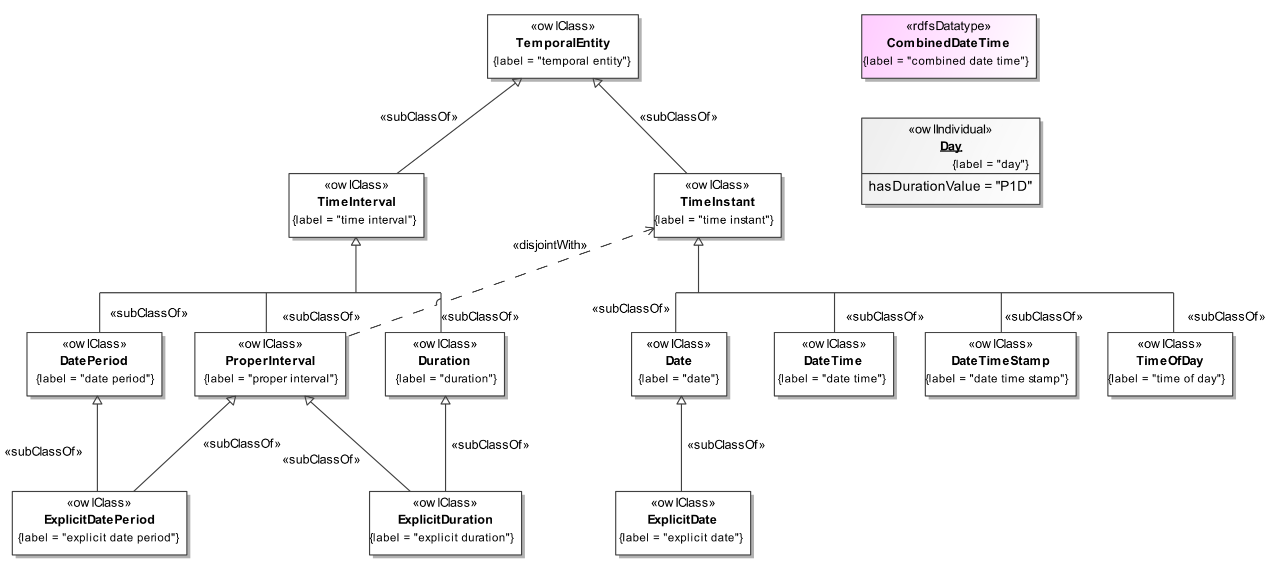

Figure 6: Class Hierarchy for the Dates and Times Ontology

The detailed annotations and axioms that comprise the Dates and Times ontology are provided in Table 8.14, below.

Table 8.14: Dates and Times Ontology Details

Classes

Name	Annotations	Class Expressions
Date (date)	Definition: calendar day on some calendar Note: A date may or may not have a value, and may be explicit or calculated. A date that has a value is one that is either explicitly set as a literal when it is created, or is some form of 'calculated date'. In an instance of date, the existence of the 'has date value' property both indicates that the date is known, and gives the value of the date. A date that does not have a value is likely one that is some form of 'calculated date', in which the actual date has not (yet) been established.	Parent Class: TimeInstant Property Restriction: ≤ 1 hasDateValue.xsd:string
DatePeriod (date period)	Definition: time span over one or more calendar days Note: A date period is defined by at least two of three properties: (1) a start date, (2) an end date, and (3) a duration. If more than one of these properties is missing, the date period may be invalid or unknown. Note: A date period is unknown if either the start date or the end date has no value. If a date period is unknown, then the duration should either be omitted or unknown (have no value).	Parent Class: TimeInterval Property Restriction: ≤ 1 hasEndDate.Date Property Restriction: ≤ 1 hasStartDate.Date Property Restriction: ≤ 1 hasDuration.Duration
DateTime (date time)	Definition: time point including a date and a time, optionally including a time zone offset Note: 'has date time value' is omitted if the 'date time' is not (yet) known. The time zone is implicitly GMT.	Parent Class: TimeInstant Property Restriction: ≤ 1 hasDateTimeValue.xsd:dateTime
DateTimeStamp (date time stamp)	Definition: time point including a date and a time that requires a time zone offset Note: 'has date time stamp value' is omitted if the 'date time stamp' is not (yet) established.	Parent Class: TimeInstant Property Restriction: ≤ 1 hasDateTimeStampValue.xsd:dateTimeSta mp
Duration (duration)	Definition: interval of time of some specific length Note: The 'has duration value' property is absent if the duration is not (yet) known.	Parent Class: TimeInterval Property Restriction: ≤ 1 hasDurationValue.xsd:string
ExplicitDate (explicit date)	Definition: date in which the 'has date value' property is required	Parent Class: Date Property Restriction: = 1 hasDateValue.xsd:string
ExplicitDatePeriod (explicit date period)	Definition: date period for which the start date, end date, and/or duration are required Note: As with 'date period', any one of {start date, end date, duration} may be omitted because the missing property can be inferred from the	Parent Class: DatePeriod, ProperInterval Property Restriction: ≤ 1 hasEndDate.ExplicitDate Property Restriction: ≤ 1 hasStartDate.ExplicitDate

	other two.	Property Restriction: ≤ 1 hasDuration.ExplicitDuration
ExplicitDuration (explicit duration)	Definition: duration for which the 'has duration value' property must have a value Note: This class is used when a duration is guaranteed to be known when it is created.	Parent Class: Duration, ProperInterval Property Restriction: = 1 hasDurationValue.xsd:string
ProperInterval (proper interval)	Definition: time interval with a non-zero extent or duration Note: Proper interval is included explicitly to enable mapping to the same term in the Time Ontology in OWL for use with the Allen intervals encoded therein. Source: https://www.w3.org/TR/owl-time/#time:ProperInterval	Parent Class: TimeInterval Class Axiom: ¬ TimeInstant
TemporalEntity (temporal entity)	Definition: time interval or instant See also: http://www.w3.org/2006/time#TemporalEntity	
TimeInstant (time instant)	Definition: temporal entity that is a member of a time scale, with no extent or duration Synonym: instant in time Synonym: time point Adapted from: https://www.omg.org/spec/DTV/ Adapted from: https://www.w3.org/TR/owl-time/#time:Instant Example: The Battle of Hastings was on '14 October 1066'. (This gives the Julian date of the battle at a granularity of 'day'. If desired, the battle could be given more precisely as a time period within that calendar day.) Note: For scales that have a granularity specified in days, a date is a time point; for scales down to the seconds, the equivalent of an xsd:dateTime or xsd:dateTimeStamp is a time point. Note: The duration of each time interval that is an instance of the time point is the granularity of the time scale of the time point.	Parent Class: TemporalEntity
TimeInterval (time interval)	Definition: segment of the time axis, a location in time, with an extent or duration Adapted from: https://www.omg.org/spec/DTV/ Adapted from: https://www.w3.org/TR/owl-time/#time:Interval Example: the day whose Gregorian calendar date is September 11, 2001 Example: the lifetime of Henry V Note: Every time interval has a beginning, an end, and a duration, even if not known. Every	Parent Class: TemporalEntity

	time interval is 'finite', a bounded segment of the time axis. The beginning or end of a time interval may be defined by reference to events that occur for a time interval that is not known. Note: Time intervals may be indefinite, meaning that their beginning is primordiality or their end is perpetuity, or both (eternity). This vocabulary assumes that indefinite time intervals exist and have some duration, but their duration is unknown.	
TimeOfDay (time of day)	Definition: explicit time, according to a clock Note: The representation similar to xsd:dateTime, but should exclude the date component and time zone. The value of the has time value property roughly corresponds to xsd:time in XML schema datatypes, which is prohibited from use in OWL due to ambiguity in its definition.	Parent Class: TimeInstant Property Restriction: = 1 hasTimeValue.xsd:string

Datatypes

Name	Annotations	Class Expressions
CombinedDateTime (combined date time)	Definition: datatype that maps to several base types for dates and times Note: Valid values must use the ISO 8601 representation for a date, or the corresponding XML Schema Datatypes representation for a date and time, or date and time including the time zone. Scope Note: There are many cases where the representation of a date may or may not include a time, and where the underlying data representation varies. This composite datatype should only be used in cases where a standard representation using one of the options in the union for date or date and time value specification does not work.	Equivalent Datatype: ∪ (xsd:string, xsd:dateTime, xsd:dateTimeStamp)

Individuals

Name	Annotations	Individual Axioms
Day (day)	Definition: explicit period of 24 hours	Type: ExplicitDuration hasDurationValue = 'P1D'

Properties

Name	Annotations	Property Axioms
hasDate (has date)	Definition: identifies a calendar day, month and year	Range: Date
hasDatePeriod (has date period)	Definition: identifies a specific window of time, including a start date, end date and/or duration	Range: DatePeriod
hasDateTime (has date time)	Definition: identifies a specific date and time of day, possibly excluding the time zone	Range: DateTime
hasDateTimeStamp (has date time stamp)	Definition: identifies a specific date and time of day, explicitly including the time zone	Range: DateTimeStamp
hasDateTimeStampValue (has date time stamp value)	Definition: specifies an actual literal (explicit) date and time, including the time zone	Range: xsd:dateTimeStamp
hasDateTimeValue (has date time value)	Definition: specifies an actual literal (explicit) date and time	Range: xsd:dateTime
hasDateValue (has date value)	Definition: specifies an actual literal (explicit) date captured in the format specified for xsd:date (i.e., ISO 8601 format), WITHOUT the time or timezone information; the semantics are identical to those of xsd:date Example: 2002-10-10 means October 10, 2002 Note: In the Finance domain, for consistency with FpML (reference FpML Coding Schemes 30 June 2014, Version 1.56, section 2.1.1), the year MUST be specified as 4 digits, and the month and day MUST be specified as 2 digits with a leading zero if needed. Times and timezones should NOT be specified.	Range: xsd:string
hasDuration (has duration)	Definition: specifies the time during which something continues Note: This duration may be omitted or unknown if either the start or end Date of the DatePeriod is an ExplicitDate.	Range: Duration
hasDurationValue (has duration value)	Definition: specifies a literal (explicit) duration (amount of time) captured in the format specified for xsd:duration (*i.e.*, ISO 8601 format); the semantics are identical to those of xsd:duration Example: -P3D means negative 3 days duration. This is used with OffsetDates to specify 3 days before (prior) to some other Date. Example: P1Y means 1 year Example: P1Y2M3DT4H5M6S means 1 year, 2 months, 3 days, 4 hours, 5 minutes, 6 seconds Example: P2M means 2 months Example: P3D means 3 days Example: PT4H means 4 hours Example: PT5M means 5 minutes Example: PT6S means 6 seconds Note: Negative durations are used to indicate relative dates that	Domain: Duration Range: xsd:string

	are before (rather than after) some other Date.	
hasEndDate (has end date)	Definition: indicates the ending date of some date period	Parent Property: hasDate Range: Date
hasExplicitDate (has explicit date)	Definition: indicates a stated date, as opposed to a calculated or unknown date, associated with something	Parent Property: hasDate Range: ExplicitDate
hasObservedDateTime (has observed date time)	Definition: indicates a date and time associated with an event, measurement, record, or observation	Range: CombinedDateTime
hasStartDate (has start date)	Definition: indicates the initial date of something	Parent Property: hasDate Range: Date
hasTimeValue (has time value)	Definition: specifies an explicit time, captured in the format specified for xsd:time (*i.e.*, ISO 8601 format), WITHOUT the date or timezone information	Range: xsd:string
precedes (precedes)	Definition: associates based on prior spatial or temporal proximity; occurs before in a logical order or sequence Source: ISO 1087 Terminology work and terminology science - Vocabulary, Second edition, 2019-09, clause 3.2.24	
succeeds (succeeds)	Definition: associates based on subsequent spatial or temporal proximity; follows in a logical order or sequence Source: ISO 1087 Terminology work and terminology science - Vocabulary, Second edition, 2019-09, clause 3.2.24	Inverse: precedes

8.8 Ontology: Designators

The designators ontology defines commonly used concepts for naming, derived in part from the patterns defined in ISO 1087 for terminology work and ISO 11179-3, Metadata Registries. It includes several very high level semiotic relationships, including defines, describes, and denotes for associating designators with the concepts they reference.

Metadata for the Designators ontology is given in Table 8.15.

Table 8.15: Designators Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/Designators/
rdfs:label	Commons Designators Ontology
dct:abstract	The designators ontology defines commonly used concepts for naming, derived in part from the patterns defined in ISO 1087 for terminology work and ISO 11179-3, Metadata Registries. It includes several very high level semiotic relationships, including defines, describes, and denotes for associating designators with the concepts they reference.

dct:contributor	Davide Sottara, Mayo Clinic
dct:contributor	Dean Allemang, Working Ontologist
dct:contributor	Elisa Kendall, Thematix Partners LLC
dct:contributor	Pete Rivett, agnos.ai
cmns-av:copyright	Copyright (c) 2014-2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2021-2022 Mayo Clinic
cmns-av:copyright	Copyright (c) 2021-2022 Working Ontologist LLC
cmns-av:copyright	Copyright (c) 2021-2022 agnos.ai U.K. Ltd
cmns-av:copyright	Copyright (c) 2022 Object Management Group, Inc.
dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/Designators/
skos:note	The designators ontology conforms with the OWL 2 DL semantics, and is outside of OWL 2 RL due to the inclusion of one minimum cardinality constraint (which is tyically ignored, but is important - see note on the Designator class) and two value restrictions. These constraints can be removed if required to support OWL RL rule-based applications that cannot be extended to support them.

An overview of the Designators ontology is given in Figure 7.

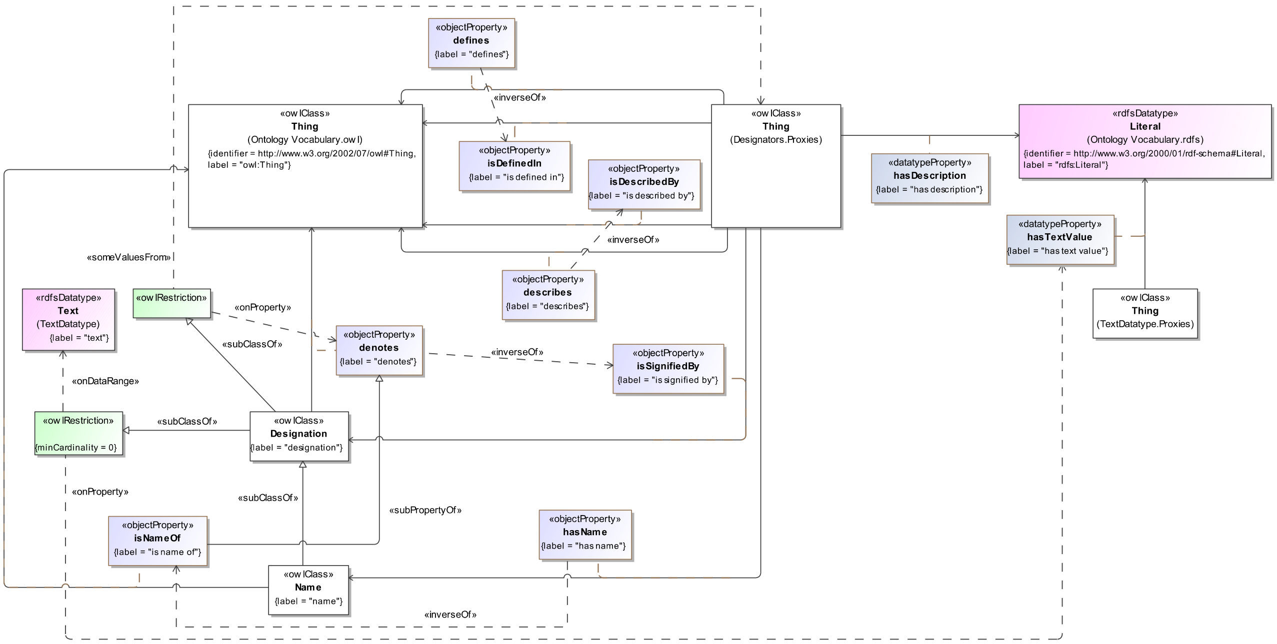

Figure 7: Overview of the Designators Ontology

The detailed annotations and axioms that comprise the Designators ontology are provided in Table 8.16, below.

Table 8.16: Designators Ontology Details

Classes

Name	Annotations	Class Expressions
Designation (designation)	<u>Definition</u>: representation for someone or something by a sign that denotes it <u>Note</u>: A designation can be a term including appellations, a proper name, or a symbol. <u>Note</u>: A designation can be linguistic or non-linguistic. It can consist of various types of characters, but also punctuation marks such as hyphens and parentheses, governed by domain-, subject-, or language-specific conventions. <u>Note</u>: Note that the use of the min 0 cardinality restriction in the definition of this class is provided as a reminder that designators are expected, in many cases, to have a text value associated with them. There are cases where this is not true, however, including symbols. And, there may be cases where the value is not known. Additionally, not all tools support rdf:langString, thus its use in the definition of the Text datatype may cause errors, for example in value and some number restrictions. Min 0 cardinality constraints are ignored by reasoners and other processors, so this allows us to say that the possible values for this property are likely either xsd:string or rdf:langString, but does not require it depending on the environment in which the ontology is	<u>Property Restriction</u>: ≥ 0 cmns-txt:hasTextValue.cmns-txt:Text <u>Property Restriction</u>: ∃ denotes.owl:Thing

	deployed. Synonym: designator Source: ISO 1087 Terminology work and terminology science - Vocabulary, Second edition, 2019-09, clause 3.4.1	
Name (name)	Definition: distinctive designation for an individual (person, organization or thing) Source: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Registry metamodel and basic attributes, Third edition, 2013-02-15	Parent Class: Designation Property Restriction: ∃ isNameOf.owl:Thing

Properties

Name	Annotations	Property Axioms
defines (defines)	Definition: specifies the meaning of something in terms of one or more of its essential qualities Note: A quality is an elementary characteristic of something. An 'essential quality' is one that provides a necessary criteria for being that thing and differentiating criteria for not being something else. See also: https://plato.stanford.edu/entries/definitions/	Inverse: isDefinedIn
denotes (denotes)	Definition: serves as a sign for something Note: Note that in some references, such as the semiotics ontology from Ontology Design Patterns, 'denotes' can be used to talk about, *e.g.*, entities denoted by proper nouns: the proper noun 'Leonardo da Vinci' denotes the person Leonardo da Vinci; as well as to talk about sets of entities that can be described by a common noun: the common noun 'person' denotes the collection of all persons in a domain of discourse. Other references that may be useful for interpreting 'denotes' include OntoLex. The interpretation of 'denotes' in this context is more general, but intended to reflect its usage in the semiotic triangle. See also: http://www.ontologydesignpatterns.org/cp/owl/semiotics.owl# See also: https://www.w3.org/2016/05/ontolex/	Domain: Designation
describes (describes)	Definition: conveys the nature of	Inverse: isDescribedBy
hasDescription (has description)	Definition: provides a textual statement, picture in words, or account that describes something Note: Note that the hasDescription property defined herein has an implicit range of rdfs:Literal. This is purposeful, so that users can specify any element that has a name with or without a language tag without concern for conflicting datatypes (*i.e.*, xsd:string vs. rdf:langString, which are logically disjoint).	Parent Property: cmns-txt:hasTextValue
hasName (has name)	Definition: is known by	Parent Property: isSignifiedBy

		Range: Name Inverse: isNameOf
isDefinedIn (is defined in)	Definition: indicates something that specifies the meaning associated with the subject Note: Typically, a concept, such as a classifier or identifier, will be defined in terms of a scheme, contract, specification, standard, or other reference.	
isDescribedBy (is described by)	Definition: has general nature or description of	
isNameOf (is name of)	Definition: denotes in some context	Parent Property: denotes Domain: Name
isSignifiedBy (is signified by)	Definition: has representation, denotation or sign	Range: Denotation Inverse: denotes

8.9 Ontology: Identifiers

The identifiers ontology defines commonly used concepts for describing identifiers and the identification schemes that define them, such as various national and international identifiers for legal entities, financial instruments, and the like, derived from the patterns specified in ISO 11179-3, Metadata Registries.

Metadata for the Identifiers ontology is given in Table 8.19.

Table 8.17: Identifiers Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/Identifiers/
rdfs:label	Commons Identifiers Ontology
dct:abstract	The identifiers ontology defines commonly used concepts for describing identifiers and the identification schemes that define them, such as various national and international identifiers for legal entities, financial instruments, and the like, derived from the patterns specified in ISO 11179-3, Metadata Registries.
dct:contributor	Elisa Kendall, Thematix Partners LLC
dct:contributor	Evan Wallace, U.S. National Institute of Standards and Technology (NIST)
dct:contributor	Pete Rivett, agnos.ai
cmns-av:copyright	Copyright (c) 2014-2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2021-2022 agnos.ai U.K. Ltd
cmns-av:copyright	Copyright (c) 2021-2022 EDM Council, Inc.

cmns-av:copyright	Copyright (c) 2021-2022 Object Management Group, Inc.
dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/Identifiers/

An overview of the Identifiers ontology is given in Figure 9.

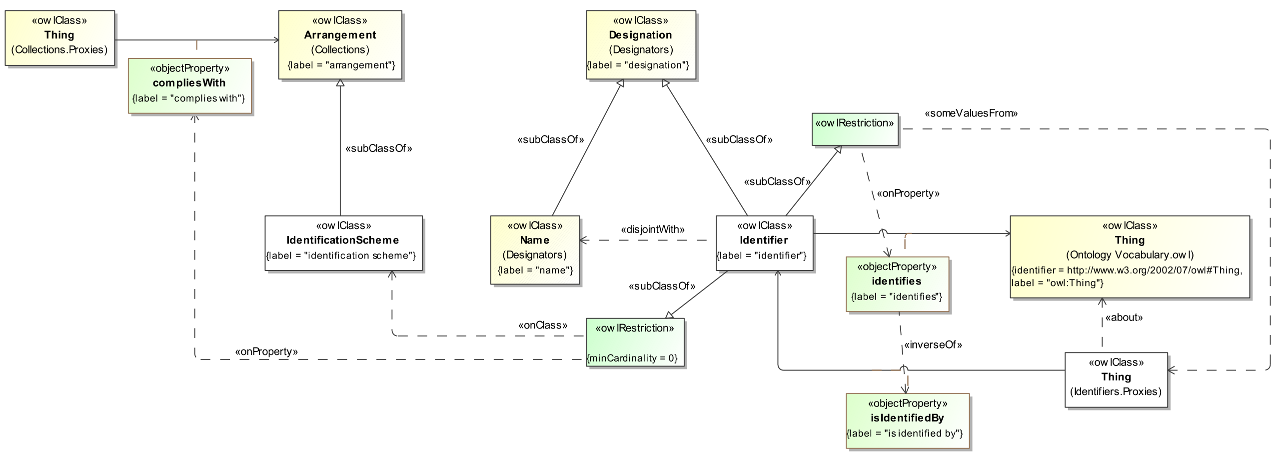

Figure 8: Overview of the Identifiers Ontology

The detailed annotations and axioms that comprise the Identifiers ontology are provided in Table 8.20, below.

Table 8.18: Identifiers Ontology Details

Classes

Name	Annotations	Class Expressions
IdentificationScheme (identification scheme)	Definition: system for minting identifiers for things that specifies constraints on the structure of the identifier Adapted from: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	Parent Class: cmns-col:Arrangement
Identifier (identifier)	Definition: sequence of characters uniquely identifying that with which it is associated Note: Note that some identifiers may be reused, or may be components of other identifiers, thus the restriction on what an identifier identifies is a 'some values' restriction rather than an exact cardinality. Examples of reusable identifiers include ticker symbols, and in the United States, vehicle license numbers, such as vanity plates that can be reassigned and moved from one car to another. Narrower constraints can be added to specific kinds of identifiers that are not reassignable and that identify exactly one thing,	Parent Class: cmns-dsg:Designation Property Restriction: ≥ 0 cmns-col:compliesWith.IdentificationScheme Property Restriction: ∃ identifies.owl:Thing Class Axiom: ¬ cmns-dsg:Name

	such as many national identifiers for people including passport numbers and, in the United States, social security numbers. Also, not all identifiers are explicitly defined in formal schemes, although they may be created or generated according to some formula. Source: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15, clause 3.1.1	

Properties

Name	Annotations	Property Axioms
identifies (identifies)	Definition: recognizes or establishes identity within some context	Parent Property: cmns-dsg:denotes Domain: Identifier
isIdentifiedBy (is identified by)	Definition: has an identifier that is unique within some context	Parent Property: cmns-dsg:isSignifiedBy Range: Identifier Inverse: identifiers

8.10 Ontology: Mapping Dates and Times to OWL Time

This ontology maps the Commons Dates and Times ontology to the widely used W3C Time Ontology in OWL recommendation, available at https://www.w3.org/TR/owl-time/. Note that users of this mapping need to be aware of datatypes that are not allowed in RDFS or OWL in the W3C Time ontology. Usage of this mapping enables use of the Allen intervals defined in the W3C ontology, however, which are useful for a number of applications.

Metadata for the Mapping Dates and Times to OWL Time ontology is given in Table 8.21.

Table 8.19: Mapping Dates and Times to OWL Time Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/MappingDatesAndTimesToOWLTime/
rdfs:label	Commons Mapping Dates and Times to OWL Time Ontology
dct:abstract	This ontology maps the Commons Dates and Times ontology to the widely used W3C Time Ontology in OWL recommendation, available at https://www.w3.org/TR/owl-time/. Note that users of this mapping need to be aware of the usage of datatypes that are not allowed in RDFS or OWL in the W3C Time ontology. Usage of this mapping enables use of the Allen intervals defined in the W3C ontology, however, which are useful for a number of applications.

dct:contributor	Elisa Kendall, Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2021-2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2022 Object Management Group, Inc.
dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/MappingDatesAndTimesToOWLTime/

The detailed annotations and axioms that comprise the Mapping Dates and Times to OWL Time ontology are provided in Table 8.22, below.

Table 8.20: Mapping Dates and Times to OWL Time Ontology Details

Classes

Name	Annotations	Class Expressions
cmns-dt:Duration		Equivalent Class: time:TemporalDuration
cmns-dt:ExplicitDate		Parent Class: time:GeneralDateTimeDescription Property Restriction: = 1 time:year Property Restriction: = 1 time:month Property Restriction: = 1 time:day
cmns-dt:ProperInterval		Equivalent Class: time:ProperInterval
cmns-dt:TemporalEntity		Equivalent Class: time:TemporalEntity
cmns-dt:TimeInstant		Equivalent Class: time:Instant
cmns-dt:TimeInterval		Equivalent Class: time:Interval

Properties

Name	Annotations	Property Axioms
time:hasXSDDuration		Parent Property: cmns-dt:hasDurationValue
time:inXSDDateTimeStamp		Parent Property: cmns-dt:hasDateTimeStampValue
time:inXSDDate		Parent Property: cmns-dt:hasDateValue

8.11 Ontology: Text Datatype

The text datatype ontology defines a custom datatype that combines language tagged and plain string values. This text datatype is useful in cases where it is not clear whether string values will be tagged or not, but where it is anticipated that multilingual strings might be appropriate.

Metadata for the Text Datatype ontology is given in Table 8.27.

Table 8.21: Text Datatype Ontology Metadata

Metadata Term	Value
OntologyIRI	https://www.omg.org/spec/Commons/TextDatatype/
rdfs:label	Commons Text Datatype Ontology
dct:abstract	The text datatype ontology defines a custom datatype that combines language tagged and plain string values. This text datatype is useful in cases where it is not clear whether string values will be tagged or not, but where it is anticipated that multilingual strings might be appropriate.
dct:contributor	Elisa Kendall, Thematix Partners LLC
dct:contributor	Evren Sirin, Stardog Union
cmns-av:copyright	2020-2022 Stardog Union
cmns-av:copyright	Copyright (c) 2020-2022 Thematix Partners LLC
cmns-av:copyright	Copyright (c) 2022 Object Management Group, Inc.
dct:license	http://opensource.org/licenses/MIT
owl:versionIRI	https://www.omg.org/spec/Commons/20220501/TextDatatype/
skos:note	Note that custom datatypes are outside the OWL 2 RL profile and so its usage in applications may need to be commented out.

An overview of the Text Datatype ontology is given in Figure 12.

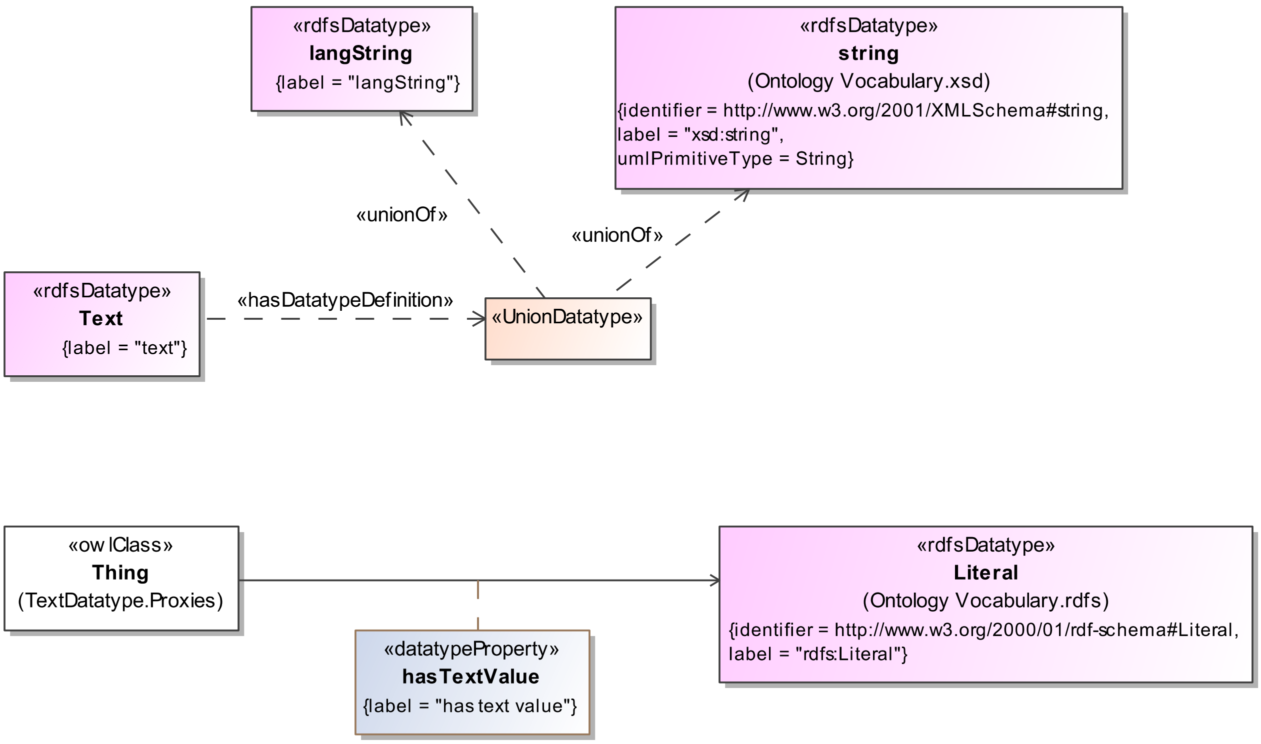

Figure 9: Overview of the Text Datatype Ontology

The detailed annotations and axioms that comprise the Text Datatype ontology are provided in Table 8.28, below.

Table 8.22: Text Datatype Ontology Details

Datatypes

Name	Annotations	Class Expressions
rdf;langString (langString)	Definition: literal with a non-empty language tag Note: This datatype declaration is included to support language-tagged strings, as defined in RDF 1.1. The rdf:langString datatype has not been incorporated directly in OWL 2 to date, and so it must be declared in order to enable its inclusion in the declaration of the Text datatype. Language-tagged strings must be well-formed according to section 2.2.9 of [BCP47]. Source: BCP 47: Tags for Identifying Languages, available at https://tools.ietf.org/search/bcp47 Source: https://www.w3.org/TR/rdf11-concepts/#section-Datatypes	
Text (text)	Definition: datatype that maps to xsd:string and rdf:langString base types for string-valued data	Equivalent Datatype: ∪ (xsd:string, rdf:langString)

	properties and annotations <u>Note</u>: Text is data in the form of characters, symbols, words, phrases, paragraphs, sentences, tables, or other character arrangements, intended to convey a meaning, and whose interpretation is essentially based upon the reader's knowledge of some natural language or artificial language. <u>Note</u>: There are cases where the representation of certain features of something, such as a name, which might be multilingual or might not, defaults to rdfs:Literal when left unspecified, although it should be limited to plain strings or language-typed strings (*i.e.*, exclude numbers, binary types, and so forth). There is no combined datatype available in RDF or OWL, however, which is the role that this datatype is intended to fulfill. <u>Scope note</u>: This composite datatype should be used in cases where a standard representation using one of the options in the union for string values does not work. Note that certain tools may not support rdf:langString, including, but not limited to some versions of Protege, and that custom datatypes are not supported in OWL 2 RL so it may need to be ignored or commented out in OWL 2 RL applications. <u>Source</u>: ISO/IEC 11179-3 Information technology - Metadata registries (MDR) - Part 3: Registry metamodel and basic attributes, Third edition, 2013-02-15	

Properties

Name	Annotations	Property Axioms
hasTextValue (has text value)	<u>Definition</u>: provides a string value for something, with or without a language tag <u>Note</u>: Note that although the intended range for this property is Text, we have left the range undefined so that it can be used with tools that do not support rdf:langString.	

Annex A: Deliverables

(normative)

The Commons ontologies are delivered as (1) RDF/XML serialized OWL (normative and definitive), and (2) Turtle serialized OWL (normative and definitive).

Each of the ontologies included in the Commons Ontology Library makes normative reference to the DCMI Dublin Core Metadata Terms [Dublin Core] and W3C Simple Knowledge Organization System (SKOS) Recommendation [SKOS], which are not part of this specification.

The individual RDF/XML files are UTF-8 conformant XML files that are also OWL 2 compliant, and may be examined using any text editor, XML editor, or RDF or OWL editor. They have been verified for syntactic correctness via the W3C RDF Validator and pass a series of unit-level tests provided by the EDM Council in our Open Knowledge Graph Innovation Laboratory (OKG IL) that cover a range of syntactic and modeling pattern issues. They have also been checked for logical consistency using the HermiT OWL 2 reasoner from Oxford University. It is anticipated that the OWL ontologies will be dereference-able, together with technical documentation (HTML) from the OMG site.