ISO/IEC JTC 1/SC 32 N 1258

Date: 2005-04-05

REPLACES: 32N1177

ISO/IEC JTC 1/SC 32

Data Management and Interchange

Secretariat: United States of America (ANSI)
Administered by Farance, Inc. on behalf of ANSI

DOCUMENT TYPE	Working Draft Text (for information or comment)
TITLE	Third ISO/IEC WD 19763-3 Framework for Metamodel Interoperability-- Part-3 : Metamodel Framework for Ontology Registry
SOURCE	Editor - OKABE, Masao & He Keqing
PROJECT NUMBER	1.32.22.01.03.00
STATUS	For comment and review.
REFERENCES	
ACTION ID.	COM
REQUESTED ACTION	Please review for the WG 2 meeting
DUE DATE	
Number of Pages	21
LANGUAGE USED	English
DISTRIBUTION	P & L Members SC Chair WG Conveners and Secretaries

Douglas Mann, Secretariat, ISO/IEC JTC 1/SC 32

Farance, Inc *, 360 Pelissier Lake Road, Marquette, MI 49855-9678, United States of America

Telephone: +1 906-249-9275; E-mail: MannD@battelle.org
available from the JTC 1/SC 32 WebSite http://staging.jtc1sc32.org/
*Farance, Inc. administers the ISO/IEC JTC 1/SC 32 Secretariat on behalf of ANSI

ISO/IEC JTC 1/SC 32 WG2

Date: 2005-4-4

ISO/IEC WD 19763-3:2005(E)-3

ISO/IEC JTC 1/SC 32/WG 2

Secretariat: ANSI

Information technology – Framework for metamodel interoperability – Part-3 : Metamodel for ontology registration

Technologies de l'information — Cadre pour l'interopérabilité du métamodèle
Partie-3: Métamodèle pour l'enregistrement ontologieque

Warning

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard. Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type: International Standard
Document subtype:
Document stage: (20) WorkingDraft
Document language: E

Contents

Figures and tables

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this part of ISO/IECWD 19763 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 19763 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 32, *Data Management and Interchange*.

ISO/IEC 19763 consists of the following parts, under the general title *Information technology* — **Framework for metamodel interoperability**:

— Part 1: Reference model

— Part 2: Core model

— Part 3: Metamodel for ontology registration

— Part 4: Metamodel for model mapping

Introduction

Today, in the EB(E-Business) or EC(E-Commerce) through the internet, the effective interchange of business transactions or other related information across countries and cultures is an important concern for people in both IT industry and other non-IT industries.

To follow the current trends of EB or EC, industrial consortia have been engaged the standardization of domain specific business objects including business process models and software components using common modeling facilities and exchanging facilities such as UML and XML. They are very active to standardizing domain specific business process models and standard modeling constructs such as data elements, entity profiles and value domains.

Moreover, interoperation among the autonomous web based applications such as Web services is becoming important. For that purpose, ontology is a key issue. Ontology is a formal expression of the universe of discourse. A lexicon, a taxonomy, a thesaurus and a usual conceptual model such as a business process model in UML are all examples of ontology. In addition ontology includes a more complex axiomatic theory.

An effort to standardize a metamodel of ontology, described in specific languages, is underway by OMG. In addition to that, to promote the ontology-based interoperation, a unified framework for registering administrative information of ontology is indispensable.

This part of ISO/IEC 19763 intends to provide a unified framework for registering administrative information of ontology, based on the ISO/IEC 19763-2 Framework for metamodel interoperability Part-2 Core Model and using the existing standardized metamodel of ontology described in specific languages.

Trademarks: UML and OMG are the trademarks of the Object Management Group.

Information Technology–Framework for metamodel interoperability –Part 3:Metamodel for ontology registration

1 Scope

The primary purpose of the multipart standard ISO/IEC 19763 is to specify a framework for metamodel interoperability. This part of ISO/IEC 19763 specifies the metamodel that provides a facility to register administrative information of ontology.

The metamodel that this part specifies is intended to promote interoperation among application systems.

It does not specify the metamodels of ontology described in specific languages and the mapping among them. They are specified in other specifications such as ODM.

Figure-1 shows the scope of this part of ISO/IEC 19763.

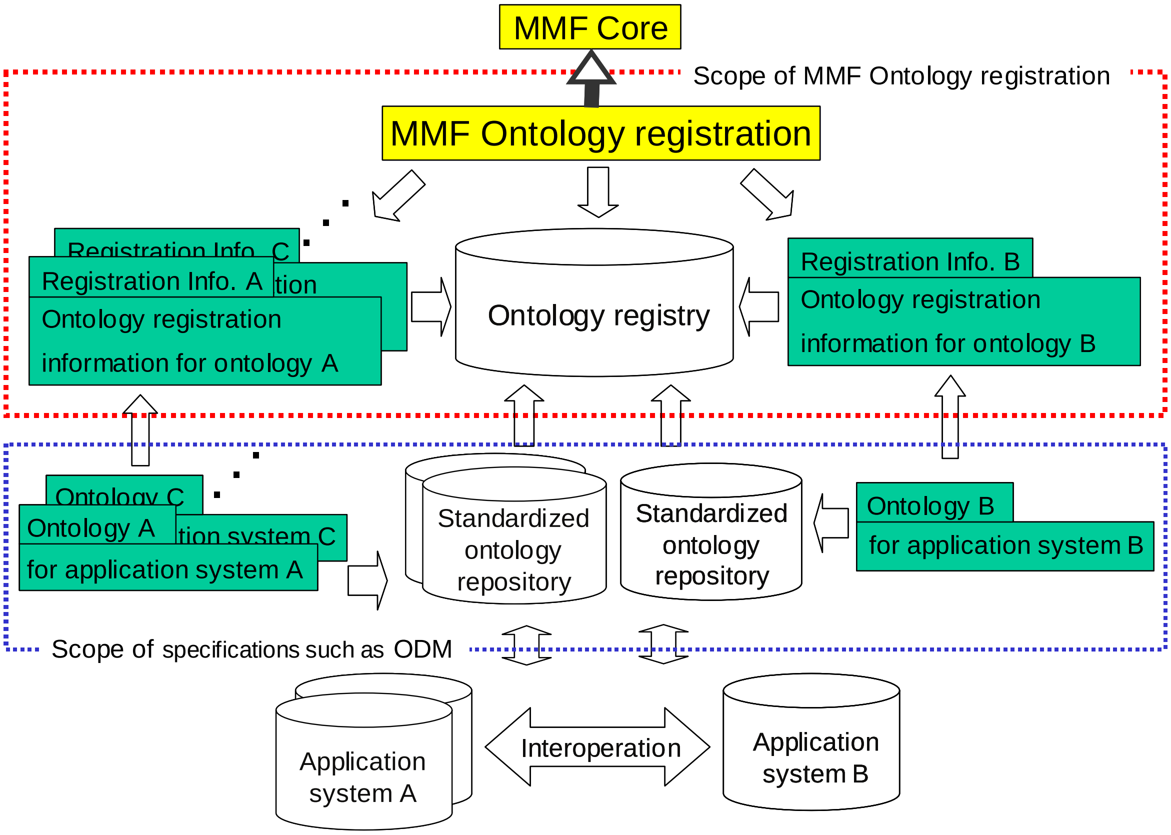


Figure 1 – Scope of MMF Ontology registration

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 11179-3, Information technology – Metadata registries (MDR) - Part 3 : Registry metamodel
ISO/IEC 19763 (all parts), Information technology – Framework for metamodel interoperability
ISO/IEC 19501-1:200x, Information technology – Unified modeling language (UML) – Part 1: Specification
ISO/IEC 19502-1:200y, Information technology – Meta Object Facility (MOF): Specification

3 Definitions and abbreviated terms

3.1 Definitions

For the purpose of this International Standard the following definitions apply.

The definitions provided in ISO/IEC 11179-3, ISO/IEC 19763 (all parts), ISO/IEC 19501-1:2002, ISO/IEC 19502-1:200x shall apply to this International Standard.

3.2 Broad terms

3.2.1 Ontology
A whole description of a universe of discourse in a formal language

3.2.2 Sentence
A piece of description that is supposed to be true in an ontology

3.2.3 Non logical symbol
The most primitive construct in an ontology that designates something in a universe of discourse.

3.3 Abbreviated terms

3.3.1 MMF Core
ISO/IEC 19763-2, Information technology – Framework for metamodel interoperability – Part-2 : Core model

3.3.2 MMF Ontology registration
ISO/IEC 19763-3, Information technology – Framework for metamodel interoperability – Part-3 : Metamodel for ontology registration

3.3.3 MDR
ISO/IEC 11179-3, Information technology – Metadata registries (MDR) - Part 3 : Registry metamodel

3.3.4 ODM
Ontology Definition Metamodel

3.3.5 URI

Uniform Resource Identifiers

4 Structure of MMF Ontology registration

4.1 Overview of MMF Ontology registration

MMF Ontology registration provides the administrative information concerning ontology registration. Figure2 shows the normative part of MMF Ontology registration metamodel.

Ontology is an abstract class that designates an ontology and has its administrative information.
Ontology_Language shows a language that describes an ontology that is designated by **Ontology.**
Ontology_Component is an abstract class that designates a sentence that composes an ontology and has its administrative information. The granularity of a sentence is not specified but is a user's choice.
Ontology_Atomic_Construct is an abstract class that designates a symbol that is used in a sentence and has its administrative information.
Reference_Ontology and **Local_Ontology** are subclasses of **Ontology**. **Reference_Ontology** designates a standardized ontology in some business domain. **Local_Ontology** designates a localized ontology for some application system based on at least one ontology that is designated by a Reference_Ontology.
Reference_Ontology_Component and **Local_Ontology_Compnent** are subclasses of **Ontology_Component**.
Reference_Ontology_Component designates a sentence that composes ontologies that are designated by **Reference_Ontology**. A sentence that is designated by **Reference_Ontology** may also compose an ontology that is designated by **Local_Ontology**. **Local_Ontology_Compnent** designates a sentence that composes one ontology that is designated by **Local_Ontology**.
Reference_Ontology_Atomic_Construct and **Local_Ontology_Atomic_Construct** are subclasses of **Ontology_Atomic_Construct. Reference_Ontology_Atomic_Construct** designates a symbol that is used in sentences that are designated by **Reference_Ontology_Component.** A symbol that designated by **Reference_Ontology** may be also used in an ontology that is designated by **Local_Ontology**.
Local_Ontology_Atomic_Construct designates a symbol that is used in sentences that are designated by **Local_Ontology_Component**. A symbol that is designated by **Local_Ontology_Atomic_Construct** can be used in only one ontology that is designated by **Local_Ontology.**

The exact specification of each metaclass is given in 4.3

4.2 Relationship between MMF Core and MMF Ontology registration

Some part of MMF Ontology registration inherits the basic structure from MMF Core. Figure 3 shows the relationship between MMF Core and MMF Ontology.

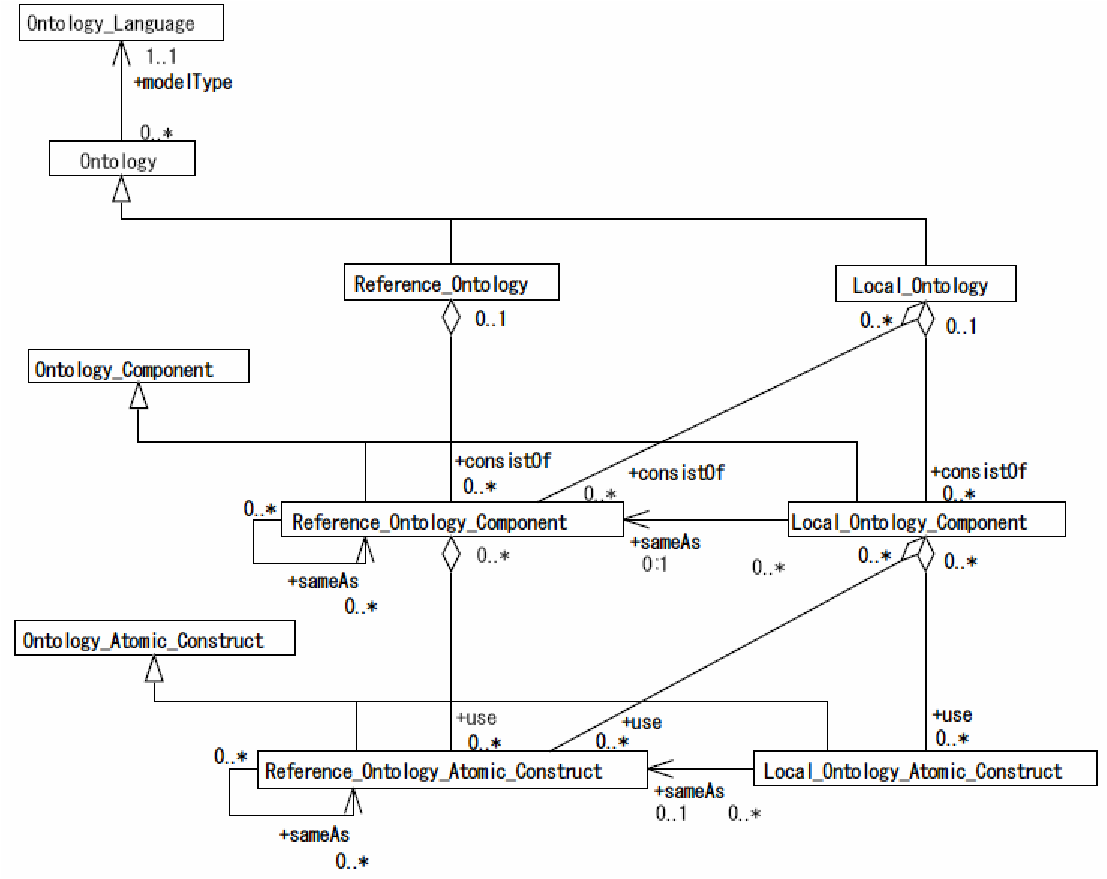


Figure 2 – Metamodel of MMF Ontology registration

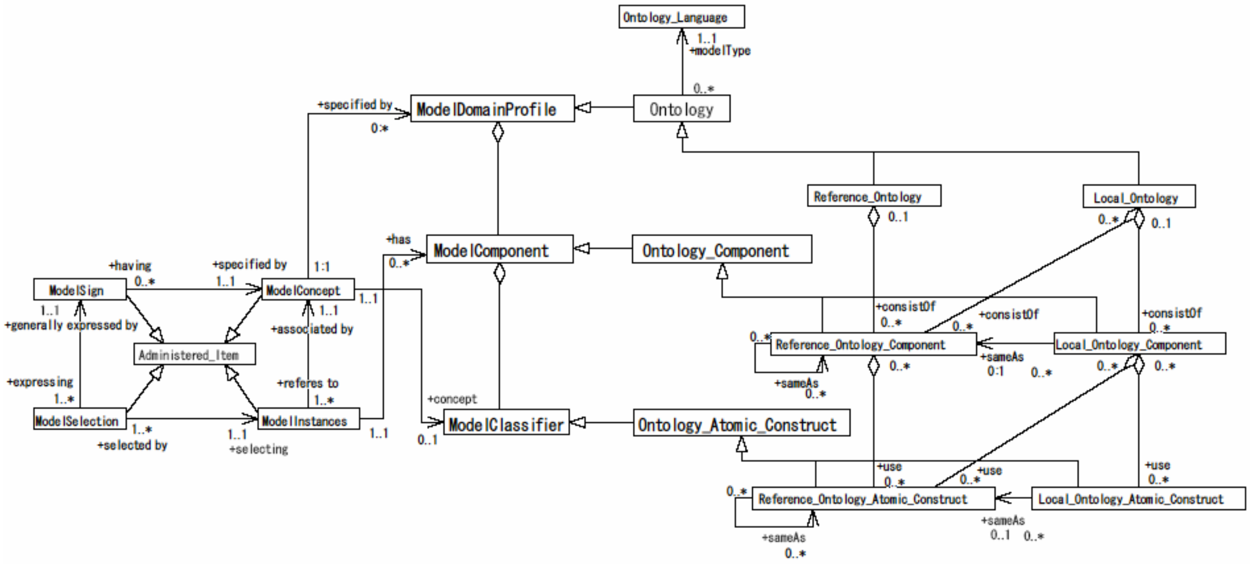


Figure 3 – Relationship between MMF Core and MMF Ontology registration

4.3 MMF Ontology registration

4.3.1 Ontology

Description			
Ontology is an abstract metaclass that is a SuperClass of Reference_Ontology and Local_Ontology.			
SuperClass	ModelDomainProfile		
Attribute	**DataType**	**Multiplicity**	**Description**
name	String	1..1	Name of the corresponding ontology
URI	String	1..1	URI where the corresponding ontology exists
Reference	**Class**	**Multiplicity**	**Description**
modelType	Ontology_Language	1..1	Ontology_Language that describes this Ontology
Constraints			
The value of attribute ‘name’ and the value of attribute ‘URI’ have to be unique in this metaclass.			

4.3.2 Reference_Ontology

Description			
Reference_Ontology is a metaclass designating a formalized ontology that is used by a community of interest.			
SuperClass	Ontology		
Reference	**Class**	**Multiplicity**	**Description**
consistOf	Reference_Ontology_Component	0..*	Reference_Ontology_Component the sentence designated by that composes the ontology designated by this Reference_Ontology
Constraints			

4.3.3 Local_Ontology

Description			
Local_Ontology is a metaclass designating an localized ontology for some application based on at least one ontology that is designated by a Reference_Ontology.			
SuperClass	Ontology		
Reference	**Class**	**Multiplicity**	**Description**
consistOf	Ontology_Component	0..*	Ontology_Component the sentence designated by that composes the ontology designated by this Local_Ontology

Constraints	

4.3.4 Ontology_Language

Description			
Ontology_Language is a metaclass that shows an ontology descriptive language.			
SuperClass			
Attribute	**DataType**	**Multiplicity**	**Description**
name	String	1..1	Name of the ontology descriptive language. It is advisable that its value be one of the values of column 'name' of Table1 at Annex C.
Constraints			
The value of attribute 'name' has to be unique in this metaclass.			

4.3.5 Ontology_Component

Description			
Ontology_Component is an abstract metaclass that is a SuperClass of Reference_Ontology_Component and Local_Ontology_Component.			
SuperClass	ModelComponent		
Attribute	**DataType**	**Multiplicity**	**Description**
namespace	String	1..1	URI where the name is uniquely identified
name	String	1..1	Name of the corresponding sentence in an ontology.
Constraints			
The value of attribute 'name' prefixed by the value of attribute 'namespace' has to be unique in this metaclass and to identify the corresponding sentence.			

4.3.6 Reference_Ontology_Component

Description			
Reference_Ontology_Component is a metaclass designating a sentence that composes an ontology that is designated by Reference_Ontology.			
SuperClass	Ontology_Component		
Reference	**Class**	**Multiplicity**	**Description**

use	Reference_Ontology_Atomic_Construct	0..*	Reference_Ontology_Atomic_Construct a non logical symbol designated by that is used by the sentence designated by this Reference_Ontology_Component
sameaAs	Reference_Ontology_Component	0..*	Reference_Ontology_Component the senetnce designated by that is interpreted exactly same as the sentence designated by this Reference_Ontology_Component . This reference is symmetric and transitive.
Constraints			

4.3.7 Local_Ontology_Component

Description			
Local_Ontology_Component is a metaclass designating a sentence that composes an ontology that is designated by Local_Ontology.			
SuperClass	Ontology_Component		
Reference	**Class**	**Multiplicity**	**Description**
use	Ontology_Atomic_Construct	0..*	Ontology_Atomic_Construct non logical symbol designated by that is used by the sentence designated by this Local_Ontology_Component
sameAs	Reference_Ontology_Component	0..1	Reference_Ontology_Component the sentence designated by that is interpreted exactly same as the sentence designated by this Local_Ontology_Component
Constraints			
Exists at most one Local_Ontology whose ‘consistOf’ is this Local_Ontology_Component			

4.3.8 Ontology_Atomic_Construct

Description			
Ontology_Atomic_Construct is an abstract metaclass that is a SuperClass of Reference_Ontology__Atomic_Construct and Local_Ontology_Atomic_Construct .			
SuperClass	ModelClassifier, Adminitered_Item		
Attribute	**DataType**	**Multiplicity**	**Description**
namespace	String	1..1	URI where the name is uniquely identified

name	String	1..1	Name of the corresponding symbol in an ontology.
Constraints			
The value of attribute ‘name’ prefixed by the value of attribute ‘namespace’ has to be unique in this metaclass.			

4.3.9 Reference_Ontology_Atomic_Construct

Description			
Reference_Ontology_Atomic_Construct is a metaclass designating an non logical symbol which is used at least one ontology which is designated by Reference_Ontology.			
SuperClass	Ontology_Atomic_Construct		
Reference	**Class**	**Multiplicity**	**Description**
sameAs	Reference_Ontology_Atomic_Construct	0..*	Reference_Ontology_Atomic_Construct the symbol designated by that is interpreted exactly same as the symbol designated by yjis Reference_Ontology_Atomic_Construct . This reference is symmetric and transitive.
Constraints			

4.3.10 Local_Ontology_Atomic_Construct

Description			
Local_Ontology_Atomic_Construct is a metaclass designating an non logical symbol which is used exactly one ontology which is designated by Local_Ontology.			
SuperClass	Ontology_Atomic_Construct		
Reference	**Class**	**Multiplicity**	**Description**
sameAs	Reference_Ontology_Atomic_Construct	0..1	Reference_Ontology_ Atomic_Construct the symbol designated by that is interpreted exactly same as the sentence designated by this Local_Ontology_Atomic_Construct
Constraints			
Exists at most one Local_Ontology whose ‘cosisitOf’ is the Local_Ontology_Component whose ‘use’ is this Local_Ontology_Atomic_Construct.			

Annex A (informative) Example of MMF Ontology registration

A.1 Example of a Reference_Ontology

Suppose that some standardization organization establishes a standardized ontology called ‘RO1’ that means ‘any buyer has a creditrating. Anthony is a buyer. Credit-A is a creditrating.’
If RO1 is described in OWL, RO1 looks as follows;

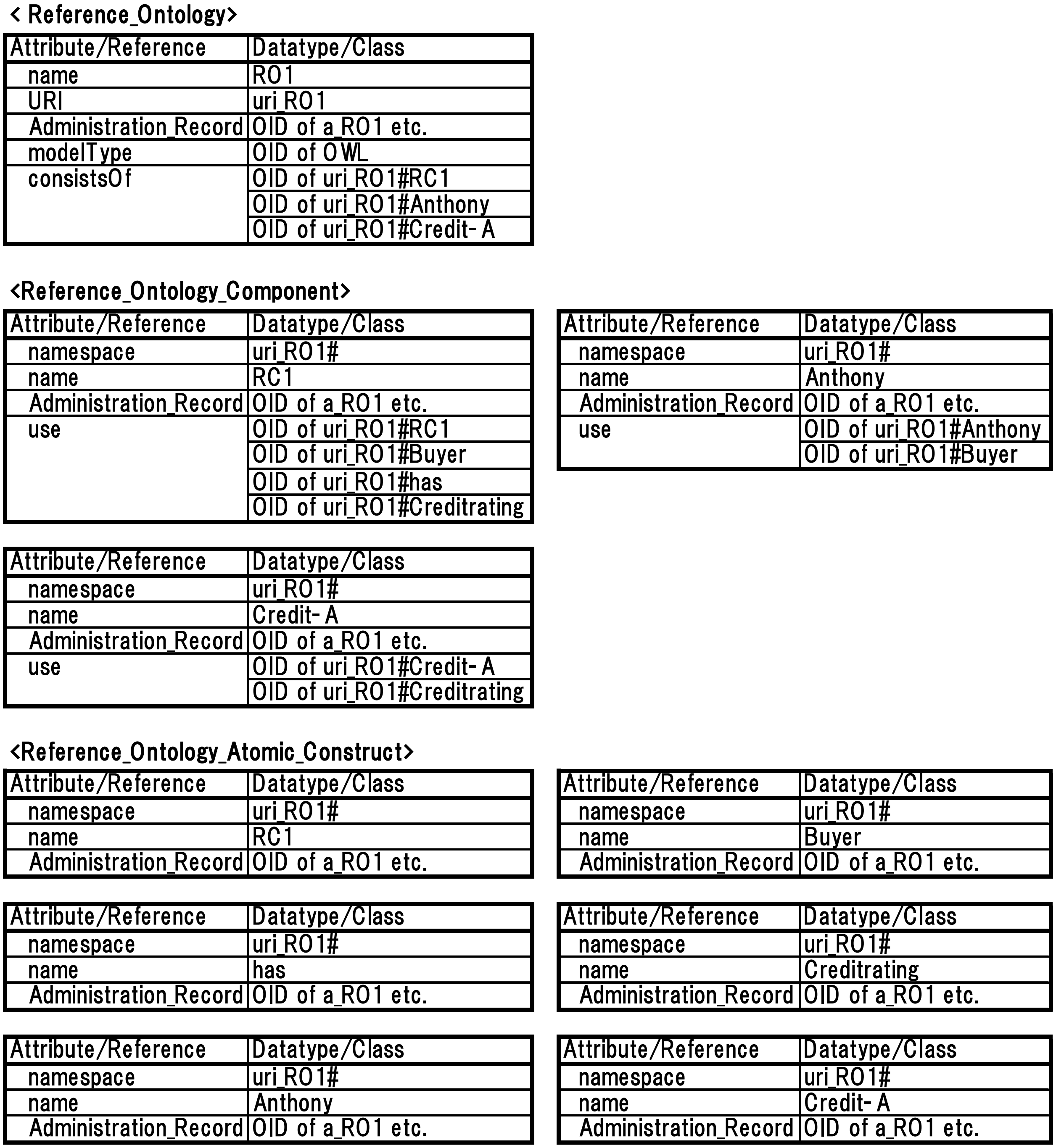
```
<rdf:Statement rdf:ID="RC1">
  <rdf:subject>
    <owl:Class rdf:about="#Buyer"/>
  </rdf:subject>
  <rdf:predicate rdf:resource=rdfs:subClassOf/>
  <rdf:object>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#has"/>
      <owl:someValueFrom rdf:resource="#Creditrating"/>
    </owl:Restriction>
  </rdf:object>
</rdf:Statement>

<rdf:Description rdf:ID="Anthony">
  <rdf:type rdf:resource="#Buyer"/>
</rdf:Description>

<rdf:Description rdf:ID="Credit-A">
  <rdf:type rdf:resource="#Creditrating"/>
</rdf:Description>
```

Note: Syntax is simplified and may not conform to OWL precisely.

Figure4 shows registrations of RO1 to MMF Ontology Registration by a_RO1 of the authority of RO1.



< Reference_Ontology>

Attribute/Reference	Datatype/Class
name	RO1
URI	uri_RO1
Administration_Record	OID of a_RO1 etc.
modelType	OID of OWL
consistsOf	OID of uri_RO1#RC1
	OID of uri_RO1#Anthony
	OID of uri_RO1#Credit-A

<Reference_Ontology_Component>

Attribute/Reference	Datatype/Class
namespace	uri_RO1#
name	RC1
Administration_Record	OID of a_RO1 etc.
use	OID of uri_RO1#RC1
	OID of uri_RO1#Buyer
	OID of uri_RO1#has
	OID of uri_RO1#Creditrating

Attribute/Reference	Datatype/Class
namespace	uri_RO1#
name	Anthony
Administration_Record	OID of a_RO1 etc.
use	OID of uri_RO1#Anthony
	OID of uri_RO1#Buyer

Attribute/Reference	Datatype/Class
namespace	uri_RO1#
name	Credit-A
Administration_Record	OID of a_RO1 etc.
use	OID of uri_RO1#Credit-A
	OID of uri_RO1#Creditrating

<Reference_Ontology_Atomic_Construct>

Attribute/Reference	Datatype/Class
namespace	uri_RO1#
name	RC1
Administration_Record	OID of a_RO1 etc.

Attribute/Reference	Datatype/Class
namespace	uri_RO1#
name	Buyer
Administration_Record	OID of a_RO1 etc.

Attribute/Reference	Datatype/Class
namespace	uri_RO1#
name	has
Administration_Record	OID of a_RO1 etc.

Attribute/Reference	Datatype/Class
namespace	uri_RO1#
name	Creditrating
Administration_Record	OID of a_RO1 etc.

Attribute/Reference	Datatype/Class
namespace	uri_RO1#
name	Anthony
Administration_Record	OID of a_RO1 etc.

Attribute/Reference	Datatype/Class
namespace	uri_RO1#
name	Credit-A
Administration_Record	OID of a_RO1 etc.

Figure 4 – Registrations of RO1

A.2 Example of another Reference_Ontology

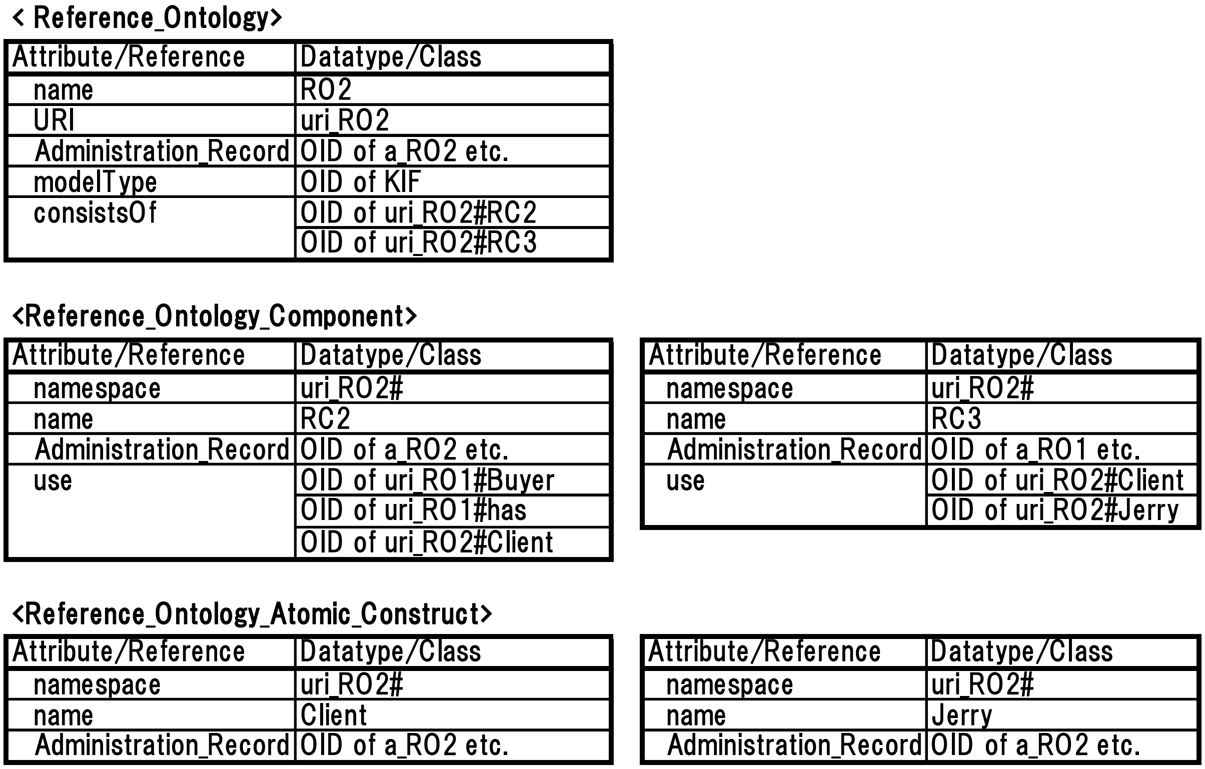
Suppose that another standardization organization establishes another standardized ontology called ‘RO2’ that means ‘any buyer has a client. Jerry is a client.’

If RO2 is described in KIF, it looks as follows.

RC2: (forall ?buyer) (implies (Buyer ?buyer) ((exist ?client) (and (has ?buyer ?client) (Client ?client))))
RC3: Client Jerry

Note: Syntax is simplified and may not conform to KIF precisely.
RC2 and RC3 are supposed to be labels that identify each sentence in RO2.
Name space of non logical symbols of RO2 is supposed to be same as RO1's.

Figure5 shows registrations of RO2 to MMF Ontology Registration by a_RO2 of the authority of RO2.
Atomic_Onto_Construct 'Buyer' and 'has' of 'RO1' are shared by RO2



< Reference_Ontology>

Attribute/Reference	Datatype/Class
name	RO2
URI	uri_RO2
Administration_Record	OID of a_RO2 etc.
modelType	OID of KIF
consistsOf	OID of uri_RO2#RC2
	OID of uri_RO2#RC3

<Reference_Ontology_Component>

Attribute/Reference	Datatype/Class
namespace	uri_RO2#
name	RC2
Administration_Record	OID of a_RO2 etc.
use	OID of uri_RO1#Buyer
	OID of uri_RO1#has
	OID of uri_RO2#Client

Attribute/Reference	Datatype/Class
namespace	uri_RO2#
name	RC3
Administration_Record	OID of a_RO1 etc.
use	OID of uri_RO2#Client
	OID of uri_RO2#Jerry

<Reference_Ontology_Atomic_Construct>

Attribute/Reference	Datatype/Class
namespace	uri_RO2#
name	Client
Administration_Record	OID of a_RO2 etc.

Attribute/Reference	Datatype/Class
namespace	uri_RO2#
name	Jerry
Administration_Record	OID of a_RO2 etc.

Figure 5 – Registrations of RO2

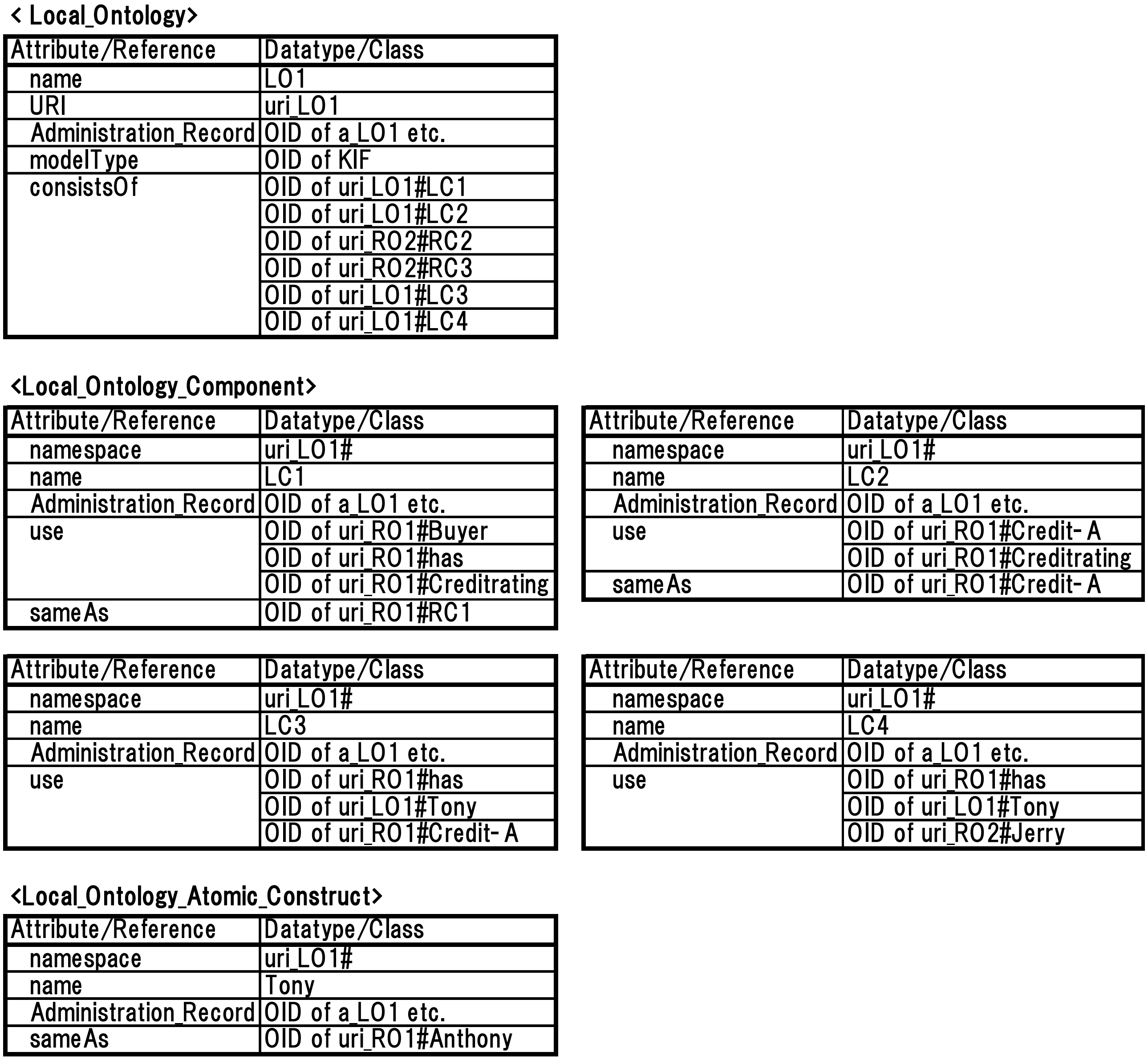
A.3 Example of a Local_Ontology

Suppose that some application system establishes its own ontology called 'LO1' based on 'RO1' and 'RO2' .
LO1 uses a knowledge 'any buyer has a Creditrating and Credit-A is a creditrating. ' of RO1 and 'any buyer has a client and Jerry is a client' of 'RO2'.
LO1 also has its own knowledge 'Tony has Credit-A and Tony has Jerry.'
Here Tony is a nickname of Anthony for this application system.
If LO1 is described in KIF, it looks as follows.
LC1: (forall ?buyer) (implies (Buyer ?buyer) ((exist ?client) (and (has ?buyer ?creditrating) (Creditrating ?client))))
LC2: Credit Credit-A
RC2: (forall ?buyer) (implies (Buyer ?buyer) ((exist ?client) (and (has ?buyer ?client) (Client ?client))))

RC3: Client Jerry
LC3: has Tony Credit-A
LC4: has Tony Jerry

Note Syntax is simplified and may not conform to KIF precisely.
LC1, LC2, LC3 and LC4 are supposed to be labels that identify each sentence in LO2.
The semantics of LC1 and LC2 are exactly same as the corresponding one in RO1.
LO1 does not have RC2 and RC3 but only refers to them in RO2.

Figure-6 shows registrations of LO1 to MMF Ontology Registration by a_LO1 of the authority of LO1.



< Local_Ontology>

Attribute/Reference	Datatype/Class
name	LO1
URI	uri_LO1
Administration_Record	OID of a_LO1 etc.
modelType	OID of KIF
consistsOf	OID of uri_LO1#LC1
	OID of uri_LO1#LC2
	OID of uri_RO2#RC2
	OID of uri_RO2#RC3
	OID of uri_LO1#LC3
	OID of uri_LO1#LC4

<Local_Ontology_Component>

Attribute/Reference	Datatype/Class
namespace	uri_LO1#
name	LC1
Administration_Record	OID of a_LO1 etc.
use	OID of uri_RO1#Buyer
	OID of uri_RO1#has
	OID of uri_RO1#Creditrating
sameAs	OID of uri_RO1#RC1

Attribute/Reference	Datatype/Class
namespace	uri_LO1#
name	LC2
Administration_Record	OID of a_LO1 etc.
use	OID of uri_RO1#Credit-A
	OID of uri_RO1#Creditrating
sameAs	OID of uri_RO1#Credit-A

Attribute/Reference	Datatype/Class
namespace	uri_LO1#
name	LC3
Administration_Record	OID of a_LO1 etc.
use	OID of uri_RO1#has
	OID of uri_LO1#Tony
	OID of uri_RO1#Credit-A

Attribute/Reference	Datatype/Class
namespace	uri_LO1#
name	LC4
Administration_Record	OID of a_LO1 etc.
use	OID of uri_RO1#has
	OID of uri_LO1#Tony
	OID of uri_RO2#Jerry

<Local_Ontology_Atomic_Construct>

Attribute/Reference	Datatype/Class
namespace	uri_LO1#
name	Tony
Administration_Record	OID of a_LO1 etc.
sameAs	OID of uri_RO1#Anthony

Figure 6 – Registrations of LO1

Annex B (informative) All metaclasses that inherit Administered_Item

Figure7 shows all metaclasses that inherit Administered_Item.

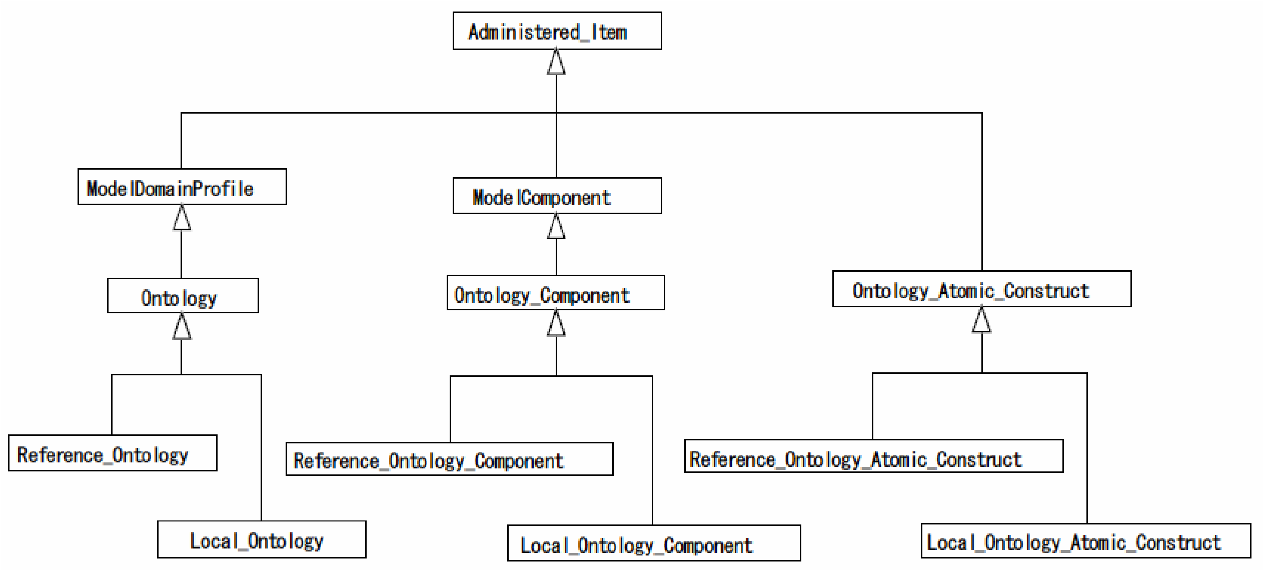
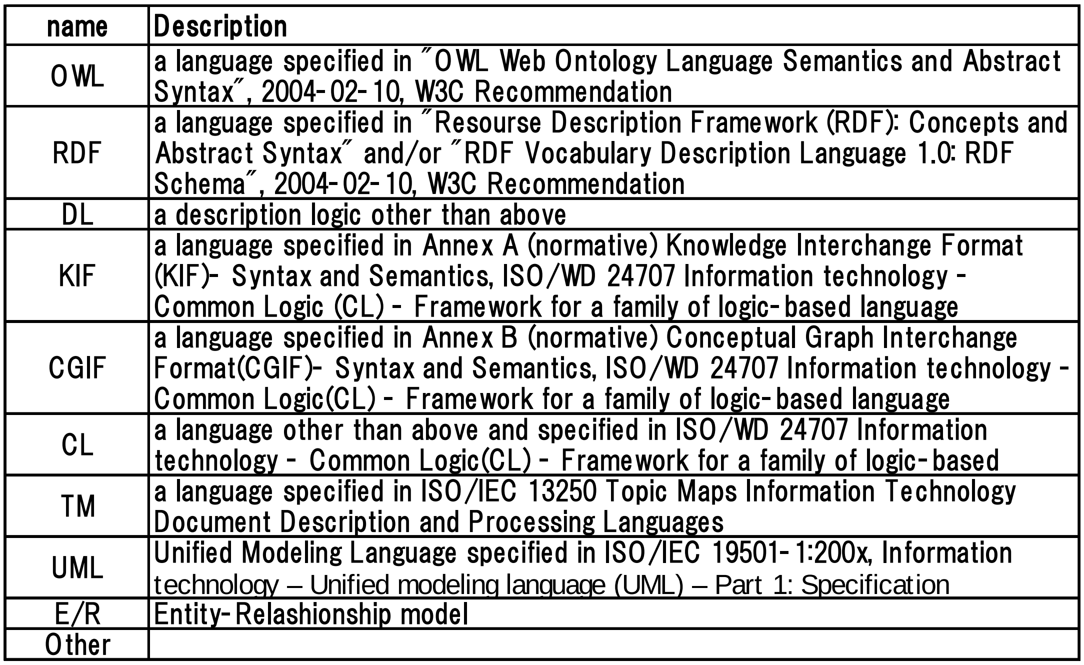


Figure 7 – All metaclasses that inherit Administered_Item

Annex C (informative) List of Ontology_Languages

It is advisable that the value of attribute ‘name’ of ‘Ontology_Language’ should be one of the values of column ‘name’ of Table1.

Table1 – List of Ontology_Languages



name	Description
OWL	a language specified in ″OWL Web Ontology Language Semantics and Abstract Syntax″, 2004-02-10, W3C Recommendation
RDF	a language specified in ″Resourse Description Framework (RDF): Concepts and Abstract Syntax″ and/or ″RDF Vocabulary Description Language 1.0: RDF Schema″, 2004-02-10, W3C Recommendation
DL	a description logic other than above
KIF	a language specified in Annex A (normative) Knowledge Interchange Format (KIF)- Syntax and Semantics, ISO/WD 24707 Information technology - Common Logic (CL) - Framework for a family of logic-based language
CGIF	a language specified in Annex B (normative) Conceptual Graph Interchange Format(CGIF)- Syntax and Semantics, ISO/WD 24707 Information technology - Common Logic(CL) - Framework for a family of logic-based language
CL	a language other than above and specified in ISO/WD 24707 Information technology - Common Logic(CL) - Framework for a family of logic-based
TM	a language specified in ISO/IEC 13250 Topic Maps Information Technology Document Description and Processing Languages
UML	Unified Modeling Language specified in ISO/IEC 19501-1:200x, Information technology – Unified modeling language (UML) – Part 1: Specification
E/R	Entity-Relashionship model
Other	

Trademarks: UML is a trademark of the Object Management Group.