

AS/NZS ISO/IEC 15476.2:2003
ISO/IEC 15476-2:2002
Reconfirmed 2014

Australian/New Zealand Standard™

**Information technology—CDIF semantic
metamodel
Part 2: Common**



Standards Australia



Document Sharing Hub

AS/NZS ISO/IEC 15476.2:2003

This Joint Australian/New Zealand Standard was prepared by Joint Technical Committee IT-015, Software Engineering. It was approved on behalf of the Council of Standards Australia on 6 May 2003 and on behalf of the Council of Standards New Zealand on 8 May 2003. It was published on 16 June 2003.

The following are represented on Committee IT-015:

Australian Computer Society
Australian Information Industry Association
Australian Society for Technical Communication (NSW)
Australian Software Metrics Association
Griffith University
New Zealand Organisation for Quality
Quality Society of Australasia
Software Engineering Australia (QLD)
Software Quality Association (ACT)
Software Quality Association (NSW)
Software Verification Research Centre
Sydney SPIN Group (Software Process Improvement Network)
Systems Engineering Society of Australia
University of New South Wales
University of South Australia
University of Technology, Sydney

Keeping Standards up-to-date

Standards are living documents which reflect progress in science, technology and systems. To maintain their currency, all Standards are periodically reviewed, and new editions are published. Between editions, amendments may be issued. Standards may also be withdrawn. It is important that readers assure themselves they are using a current Standard, which should include any amendments which may have been published since the Standard was purchased.

Detailed information about joint Australian/New Zealand Standards can be found by visiting the Standards Australia web site at www.standards.com.au or Standards New Zealand web site at www.standards.co.nz and looking up the relevant Standard in the on-line catalogue.

Alternatively, both organizations publish an annual printed Catalogue with full details of all current Standards. For more frequent listings or notification of revisions, amendments and withdrawals, Standards Australia and Standards New Zealand offer a number of update options. For information about these services, users should contact their respective national Standards organization.

We also welcome suggestions for improvement in our Standards, and especially encourage readers to notify us immediately of any apparent inaccuracies or ambiguities. Please address your comments to the Chief Executive of either Standards Australia International or Standards New Zealand at the address shown on the back cover.

This Standard was issued in draft form for comment as DR 03145.

STANDARDS AUSTRALIA/STANDARDS NEW ZEALAND

RECONFIRMATION

OF

AS/NZS ISO/IEC 15476.2:2003

Information technology—CDIF semantic metamodel

Part 2: Common

RECONFIRMATION NOTICE

Technical Committee IT-015 has reviewed the content of this publication and in accordance with Standards Australia procedures for reconfirmation, it has been determined that the publication is still valid and does not require change.

Certain documents referenced in the publication may have been amended since the original date of publication. Users are advised to ensure that they are using the latest versions of such documents as appropriate, unless advised otherwise in this Reconfirmation Notice.

Approved for reconfirmation in accordance with Standards Australia procedures for reconfirmation on 17 March 2014.

The following are represented on Technical Committee IT-015:

Australian Computer Society
Australian Society for Technical Communication (NSW)
Charles Sturt University
Department of Defence (Australia)
Griffith University
La Trobe University
National Association of Testing Authorities Australia
National ICT Australia
NSW Business Chamber
Quantitative Enterprise Software Performance
Systems Engineering Society of Australia
University of Technology, Sydney

NOTES

Australian/New Zealand Standard™

Information technology—CDIF semantic metamodel Part 2: Common

First published as AS/NZS ISO/IEC 15476.2:2003.

COPYRIGHT

© Standards Australia/Standards New Zealand

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher.

Jointly published by Standards Australia International Ltd, GPO Box 5420, Sydney, NSW 2001, and Standards New Zealand, Private Bag 2439, Wellington 6020

ISBN 0 7337 5292 6

PREFACE

This Standard was prepared by the Joint Standards Australia/Standards New Zealand Committee IT-015, Software Engineering.

This Standard is identical with, and has been reproduced from ISO/IEC 15476-2:2002, *Information technology—CDIF semantic metamodel—Part 2: Common*.

The objective of this Standard is to define the Common Subject Area of the CDIF semantic metamodel. This subject area contains meta-objects that are used as the basis of the other subject area standards, and also meta-relationships and meta-attributes that are applicable to all meta-objects.

This Standard is Part 2 of AS/NZS ISO/IEC 15476, *Information technology—CDIF semantic metamodel*, which is published in parts as follows:

Part 1: Foundation

Part 2: Common (this Standard)

As this Standard is reproduced from an international standard, the following applies:

- (a) Its number appears on the cover and title page while the international standard number appears only on the cover.
- (b) In the source text 'this International Standard' should read 'this Australian/New Zealand Standard'.
- (c) A full point substitutes for a comma when referring to a decimal marker.

References to International Standards should be replaced by references to Australian or Australian/New Zealand Standards, as follows:

<i>Reference to International Standard</i>		<i>Australian/New Zealand Standard</i>	
ISO/IEC		AS/NZS	
15474	Information technology—CDIF framework	ISO/IEC 15474	Information technology—CDIF framework
15474-1	Part 1: Overview	ISO/IEC 15474.1	Part 1: Overview
15474-2	Part 2: Modelling and extensibility	ISO/IEC 15474.2	Part 2: Modelling and extensibility
15476	Information technology—CDIF semantic metamodel	ISO/IEC 15476	Information technology—CDIF semantic metamodel
15476-1	Part 1: Foundation	ISO/IEC 15476.1	Part 1: Foundation

CONTENTS

Page

1	Scope.....	1
2	Conformance	2
2.1	Input conformance	2
2.2	Output conformance	3
2.3	Round-trip conformance	3
3	Normative references.....	3
4	Terms and definitions	4
4.1	From other standards	4
4.1.1	ISO/IEC 15474-1	4
4.1.2	ISO/IEC 13238-1	4
4.1.3	For this standard	4
5	Symbols (and abbreviated terms)	5
5.1	Naming, diagramming and definition conventions	5
5.2	Abbreviations	5
6	Common subject area overview	5
6.1	Introduction	5
6.2	Diagram	6
6.3	Classification	6
6.4	Semantic information.....	6
6.5	Presentation information.....	7
6.6	Naming	7
6.7	Audit data	8
6.8	Constraints	8
6.9	Derivation.....	8
6.10	Abstraction levels	8
6.11	General structuring mechanism	9
6.11.1	Introduction	9
6.11.2	Diagram	9
6.11.3	Decomposition and structure definitions	9
6.11.4	Accessing a specific component in a shared definition	10
6.11.5	Identical instances in shared definitions.....	10
6.11.6	Defining interfaces	11
6.11.7	Model	11
6.12	Computable languages.....	11
7	Common subject area summary	12
7.1	AttributableMetaObject classification hierarchy	12
7.2	MetaEntity summary	13
7.3	MetaRelationship summary	16
8	Common subject area specification.....	19
8.1	Introduction	19
8.2	Subject area definition.....	19
8.3	Meta-entity definitions	20
8.3.1	AbstractionLevel	20
8.3.2	AlternateName	21
8.3.3	ComponentObject	22
8.3.4	DefinitionObject.....	23
8.3.5	Derivation	25
8.3.6	EquivalenceSet.....	27
8.3.7	Model	27

8.3.8	PresentationInformationObject.....	28
8.3.9	ReferencedElement.....	29
8.3.10	RootEntity.....	29
8.3.11	SemanticInformationObject.....	30
8.3.12	TextualConstraint.....	31
8.3.13	ToolUser.....	33
8.4	Meta-relationship definitions.....	34
8.4.1	ComponentObject.IsActualFor.ComponentObject.....	34
8.4.2	ComponentObject.References.DefinitionObject.....	34
8.4.3	DefinitionObject.Contains.ComponentObject.....	35
8.4.4	DefinitionObject.ContainsAsFormal.ComponentObject.....	35
8.4.5	EquivalenceSet.HasMember.ComponentObject.....	36
8.4.6	Model.HasRoot.DefinitionObject.....	36
8.4.7	ReferencedElement.DefinesPath.ComponentObject.....	37
8.4.8	RootEntity.CreatedBy.ToolUser.....	37
8.4.9	RootEntity.Has.AlternateName.....	38
8.4.10	RootEntity.LastUpdatedBy.ToolUser.....	38
8.4.11	RootEntity.IsRelatedTo.RootEntity.....	39
8.4.12	RootEntity.Uses.AlternateName.....	39
8.4.13	SemanticInformationObject.IsCategorizedIn.AbstractionLevel.....	40
8.4.14	SemanticInformationObject.ProducedBy.Derivation.....	40
8.4.15	SemanticInformationObject.UsedIn.Derivation.....	41
8.4.16	TextualConstraint.IsConstraintOn.SemanticInformationObject.....	41

Table of Illustrations

Figure 1 – CDIF family of standards.....	1
Figure 2 – Common subject area.....	6
Figure 3 – AlternateName example.....	8
Figure 4 – General structuring mechanism.....	9
Figure 5 – Referring to a specific instance inside a shared definition.....	10

Table of Tables

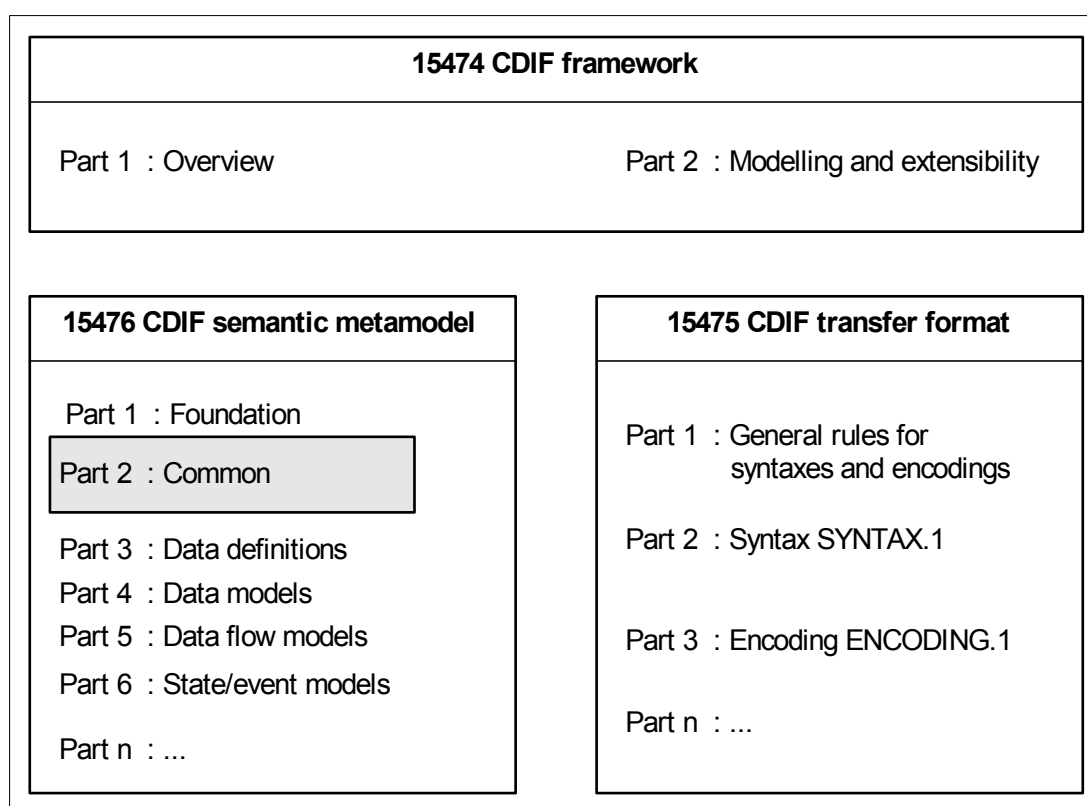
Table 1 – Allowable values for computable languages and appropriate references.....	11
---	----

AUSTRALIAN/NEW ZEALAND STANDARD

Information technology — CDIF semantic metamodel —**Part 2:
Common****1 Scope**

The CDIF family of standards is primarily designed to be used as a description of a mechanism for transferring information between modelling tools. It facilitates a successful transfer when the authors of the importing and exporting tools have nothing in common except an agreement to conform to CDIF. The language that is defined for the transfer format also has applicability as a general language for import/export from repositories. The CDIF semantic metamodel defined for CASE also has applicability as the basis of standard definitions for use in repositories.

The standards which form the complete family of CDIF standards are documented in ISO/IEC 15474-1:2002, *Information technology — CDIF framework — Part 1: Overview*. These standards cover the overall framework, the transfer format and the CDIF semantic metamodel.

**Figure 1 – CDIF family of standards**

The diagram in Figure 1 depicts the various standards that comprise the CDIF family of standards. The shaded box depicts this Standard and its position in the CDIF family of standards.

This standard defines the Common Subject Area of the CDIF semantic metamodel. This subject area contains meta-objects that are used as the basis of the other subject area standards, and also meta-relationships and meta-attributes that are applicable to all meta-objects.

This document is intended to be used by anyone wishing to understand and/or use CDIF. This document provides a definition of a single subject area of the CDIF semantic metamodel. It is suitable for:

- Those evaluating CDIF,
- Those who wish to understand the principles and concepts of a CDIF transfer, and
- Those developing importers and exporters.

This document, ISO/IEC 15474-1:2002, *Information technology — CDIF framework — Part 1: Overview*, and the Framework document ISO/IEC 15474-2:2002, *Information technology — CDIF framework — Part 2: Modelling and extensibility*, should be read first when initially exploring CDIF and before attempting to read other documents in the CDIF family of standards.

While there are no specific prerequisites for reading this document, it will be helpful for the reader to have familiarity with the following:

- Entity-Relationship-Attribute modelling;
- Modelling (CASE) tools;
- Information repositories;
- Data dictionaries;
- Multiple meta-layer modelling.

2 Conformance

A product is fully standards conformant to a CDIF subject area standard if and only if it is input-conformant, output-conformant and round-trip conformant to each and every *MetaEntity*, *MetaRelationship*, *MetaAttribute*, and *AttributableMetaObject* which is defined and/or used in that standard, and it is also CDIF architecture conformant. A product may be partially input-conformant, and/or partially output-conformant, and/or partially round-trip conformant to a CDIF subject area standard.

2.1 Input conformance

Input conformance for a specific *MetaEntity*, *MetaRelationship*, *MetaAttribute*, or *AttributableMetaObject* (short: *CollectableMetaObject*) is determined by applying the following test:

A set of meta-data containing all meanings and structures standardized by a CDIF subject area is imported by the product under test. Then the meta-data which has arrived in the product is examined. The following options exist for the relation between the input (CDIF) meta-data and the imported (product) meta-data:

For a specific *CollectableMetaObject*:

- 1 The product is input conformant if each instance of the specific *CollectableMetaObject* has arrived in the product without change of meaning or structure. If the *CollectableMetaObject* is a meta-entity or meta-relationship, its structural relationships to other *CollectableMetaObjects* have been preserved. If the *CollectableMetaObject* is a meta-attribute, the value of the meta-attribute has been preserved.
- 2 The product is input morphing conformant if each instance of the specific *CollectableMetaObject* has arrived in the product, but with some changes in meaning or structure. If the *CollectableMetaObject* is a meta-attribute, the value(s) for some instances of the meta-attribute have changed.
- 3 The product is not input conformant for that *CollectableMetaObject* if neither of the previous tests is satisfied.

2.2 Output conformance

Output conformance for a specific *CollectableMetaObject* is determined by applying the following test:

For the product being tested, a set of meta-data that includes all possible meanings and structures representable in that product is exported. Then the meta-data that has been exported is examined. The following options exist for the relation between the product's meta-data and the exported (CDIF) meta-data:

For a specific *CollectableMetaObject*:

- 1 The product is output conformant if all of the meaning and structure for the specific *CollectableMetaObject* has been represented as meta-data in the product and has been exported as one or more instances of that *CollectableMetaObject*. If the *CollectableMetaObject* is a meta-attribute, the correct value of the meta-attribute has been exported.
- 2 The product is output morphing conformant if each instance of meta-data in the product that has the same meaning and structure as the *CollectableMetaObject* has been exported, but some instances have been exported as a different *CollectableMetaObject* or some of the meaning and structure has been changed.
- 3 If the product does not represent the meaning and structure associated with the *CollectableMetaObject*, output conformance for that *CollectableMetaObject* is not applicable to the product.
- 4 In all other cases, the product is not output conformant for that *CollectableMetaObject*.

2.3 Round-trip conformance

Round-trip conformance for a specific *CollectableMetaObject* is determined by applying the following test:

A set of meta-data containing all meanings and structures standardized by a CDIF subject area is imported by the product under test. Then the meta-data is exported again. The following options exist for the relation between the input meta-data and the output meta-data:

For a specific *CollectableMetaObject*:

- 1 The product is round-trip conformant if the meaning and structure of each instance of the *CollectableMetaObject* is preserved without changes during the round-trip. For a vendor to claim round-trip conformance, it is also necessary for the tool to be able to perform create, read, update, and delete operations on the imported (product) meta-data corresponding to the instances of the *CollectableMetaObject*.
- 2 The product is round-trip morphing conformant if each instance of the input *CollectableMetaObject* is preserved, but with some changes in meaning and/or structure. If the *CollectableMetaObject* is a meta-entity or meta-relationship, some of its instances' structural relationships to other *CollectableMetaObjects* have changed, or some instances have been transformed into other *CollectableMetaObjects*, or instances of other *CollectableMetaObjects* have been transformed into instances of the *CollectableMetaObject*. If the *CollectableMetaObject* is a meta-attribute, the values of some instances of the meta-attribute have changed or the domain of the meta-attribute has changed.
- 3 In all other cases, the product is not round-trip conformant for that *CollectableMetaObject*.

3 Normative references

The following normative documents contain provisions, which, through reference in this text, constitute provisions of this part of ISO/IEC 15476. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of ISO/IEC 15476 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO/IEC 13238-1:—¹⁾, *Information technology — Data management export/import — Part 1: Standardization framework*

ISO/IEC 15474-1:2002, *Information technology — CDIF framework — Part 1: Overview*

ISO/IEC 15474-2:2002, *Information technology — CDIF framework — Part 2: Modelling and extensibility*

ISO/IEC 15476-1:2002, *Information technology — CDIF semantic metamodel — Part 1: Foundation*

4 Terms and definitions

For the purposes of this part of ISO/IEC 15476, the following definitions apply. Unless otherwise noted, the definitions are specific to this part of ISO/IEC 15476.

4.1 From other standards

4.1.1 ISO/IEC 15474-1

This part of ISO/IEC 15476 makes use of the following terms defined in ISO/IEC 15474-1:

CDIF
CDIF family of standards
CDIF semantic metamodel
CDIF meta-metamodel
CDIF transfer
Instance
Meta-attribute
Meta-entity
Metamodel
Meta-object
Meta-relationship
Model
Subject area
Transfer
Transfer format

4.1.2 ISO/IEC 13238-1

This part of ISO/IEC 15476 makes use of the following terms from ISO/IEC 13238-1:

Exporter
Importer

4.1.3 For this standard

For the purposes of this part of ISO/IEC 15476 new terms are defined when introduced. Double quotes are used to introduce new terms (e.g., “model layer”)

1) To be published.

5 Symbols (and abbreviated terms)

5.1 Naming, diagramming and definition conventions

Conventions for naming, diagramming, describing and defining meta-objects can be found in Clause 7 of the framework document (ISO/IEC 15474-2:2002, *Information technology — CDIF framework — Part 2: Modelling and extensibility*).

5.2 Abbreviations

The following abbreviation is used in this international standard:

CDIF CASE Data Interchange Format (originally)

6 Common subject area overview

6.1 Introduction

The Common subject area is intended to cover concepts such as synonyms, decomposition, and audit data, that are common or useful to all of the CDIF semantic metamodel. The Common subject area covers aspects required for all objects described in a CDIF transfer. Concepts defined here include alternate names, details of the tool user who created or updated any object, and constraints on objects.

6.2 Diagram

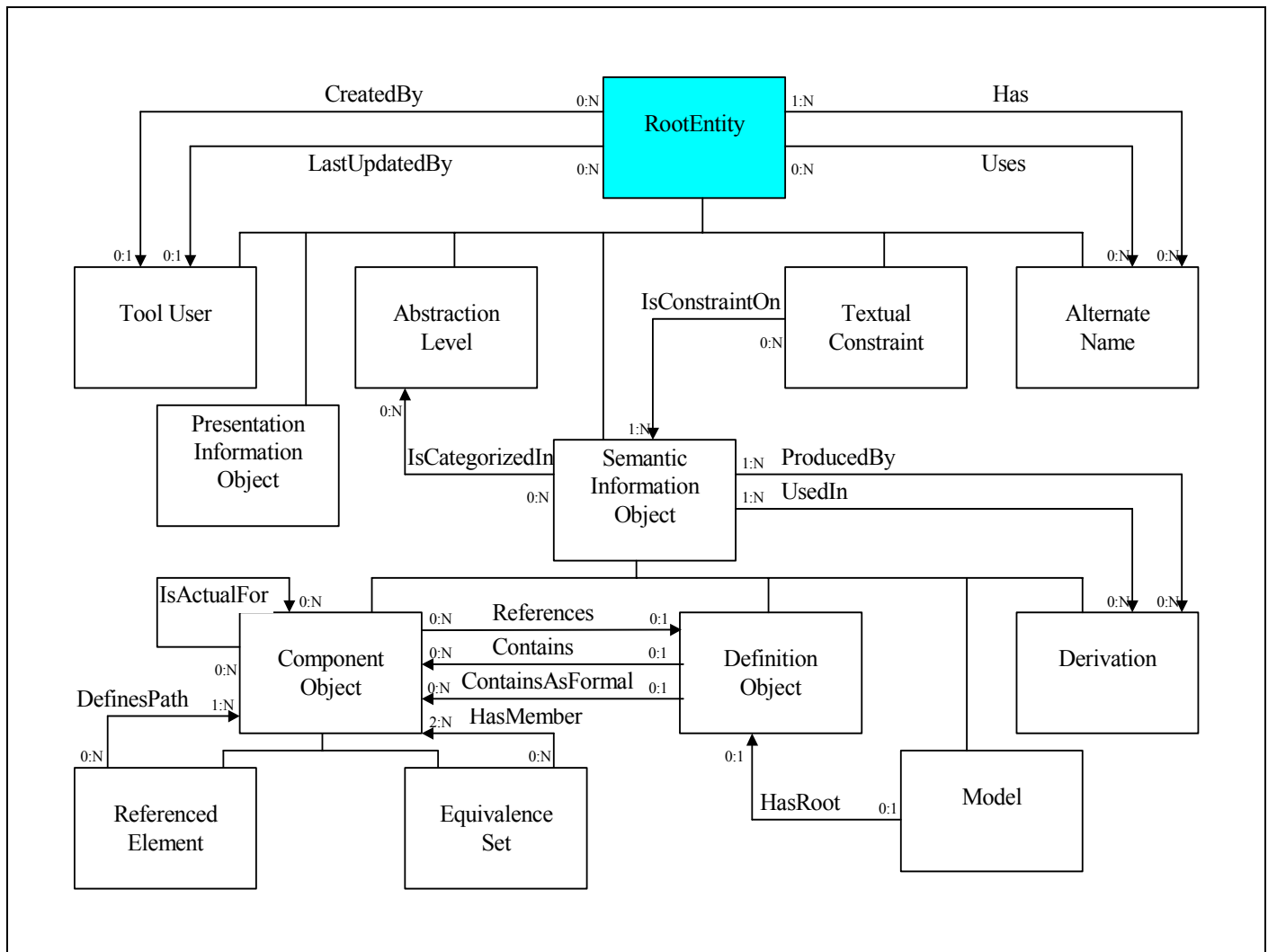


Figure 2 – Common subject area

6.3 Classification

The CDIF semantic metamodel is hierarchical in nature. The top of the semantic metamodel is *RootEntity*. *RootEntity* establishes a set of common meta-attributes and meta-relationships to be inherited by all meta-objects. *RootEntity* is an abstract supertype that should not be instantiated and shall not have any supertypes defined.

There are two main subtypes defined for *RootEntity* — *SemanticInformationObject* and *PresentationInformationObject*. The other subtypes are *AbstractionLevel*, *AlternateName*, *TextualConstraint*, and *ToolUser*. These are all described in the following sub-clauses.

6.4 Semantic information

SemanticInformationObject is the supertype of all meta-entities defined within the semantic information portion of the CDIF semantic metamodel. These describe model objects (e.g., objects in a general business model). All meta-entities that convey semantic information, defined in the set of standards forming the CDIF semantic metamodel,

other than *RootEntity* and *SemanticInformationObject*, are defined as either direct or indirect subtypes of *SemanticInformationObject*.

The meta-entity *SemanticInformationObject* is given local meta-attributes of *BriefDescription* and *FullDescription*.

6.5 Presentation information

PresentationInformationObject is the supertype for all the meta-entities defined within the presentation information portion of the CDIF semantic metamodel. These contain the graphical information about the associated semantic models. For example, the location and shape of an object in the semantic model.

6.6 Naming

The concept of a "Name" for an object varies considerably from method to method and object to object. In some cases, an object can have a short name, a long name and alternates or synonyms for each. In other cases, no distinct name is required. Due to these differences, no *Name* meta-attributes have been provided at a high level in the *AttributableMetaObject* hierarchy to be automatically inherited. Name meta-attributes have been added specifically as required. These meta-attributes have been called *Name* and *LongName* when used.

Often objects can be given alternate names, or synonyms. These are optional, and there may be more than one provided. To cater for this in a standard manner, an *AlternateName* meta-entity has been provided which can be related to any instance of *RootEntity* and its other subtypes, except *AlternateName*. This provides a simple way of defining such alternates, but it shall only be used where either *Name*, *LongName*, or both, are provided as meta-attributes of the meta-entity to which it is related. Alternatives for *Name* and *LongName* are recorded in the *OtherName* and *OtherLongName* meta-attributes of *AlternateName*, respectively. *AlternateName* shall not be used to provide a name for an object where no *Name* or *LongName* meta-attribute has been provided since a name is only **alternate** where there is one already defined. When it is necessary to add a name to an unnamed object, such naming should be done through the use of extensibility by adding an explicit *Name* meta-attribute.

Consider a data type called **CustomerNumber**; it may also be known by other names in the system being modeled, including **CustomerNo** and **CustNo**. These are synonyms for the same data type. The name **CustomerNumber** would be held in the *Name* meta-attribute of the instance of *DataType*. Each of the synonyms would be held in a separate instance of *AlternateName* in the *OtherName* meta-attribute. These two instances of *AlternateName* would be related to the instance of *DataType* by the meta-relationship *RootEntity.Has.AlternateName*. This set of meta-objects records that the data type has two synonyms.

Now consider how to record which synonym is being used in any particular context - this is where the meta-relationship *RootEntity.Uses.AlternateName* is required. The following example, where different organizations refer to vendors as either Vendors, Contractors, or Suppliers, shows how the CDIF semantic metamodel supports this concept. Assume that the part of a subject area dealing with contracts includes the *Organization* and *Resource* meta-entities and a meta-relationship called *Organization.Utilizes.Resource*. Assume, also, that the **Vendor** instance of *Resource* has two *AlternateNames*: **Supplier** and **Contractor**. If an organization, say Reprographics, uses the term **Vendor** to refer to those it relies on for service, this would be recorded with an instance of *Organization.Utilizes.Resource* from the **Reprographics** instance of *Organization* to the **Vendor** instance of *Resource*. If another organization, FoodServices, refers to the vendor for cafeteria operations as a **Supplier**, this would be recorded via an instance of *Organization.Utilizes.Resource* relating the **FoodServices** instance of *Organization* to the **Vendor** instance of *Resource*. But an instance of *RootEntity.Uses.AlternateName* between **FoodServices** and **Supplier** is used to indicate that the **Supplier** *AlternateName* is what FoodServices calls vendors. Figure 3 shows both the subject area fragment and a depiction of the instances discussed in this example.

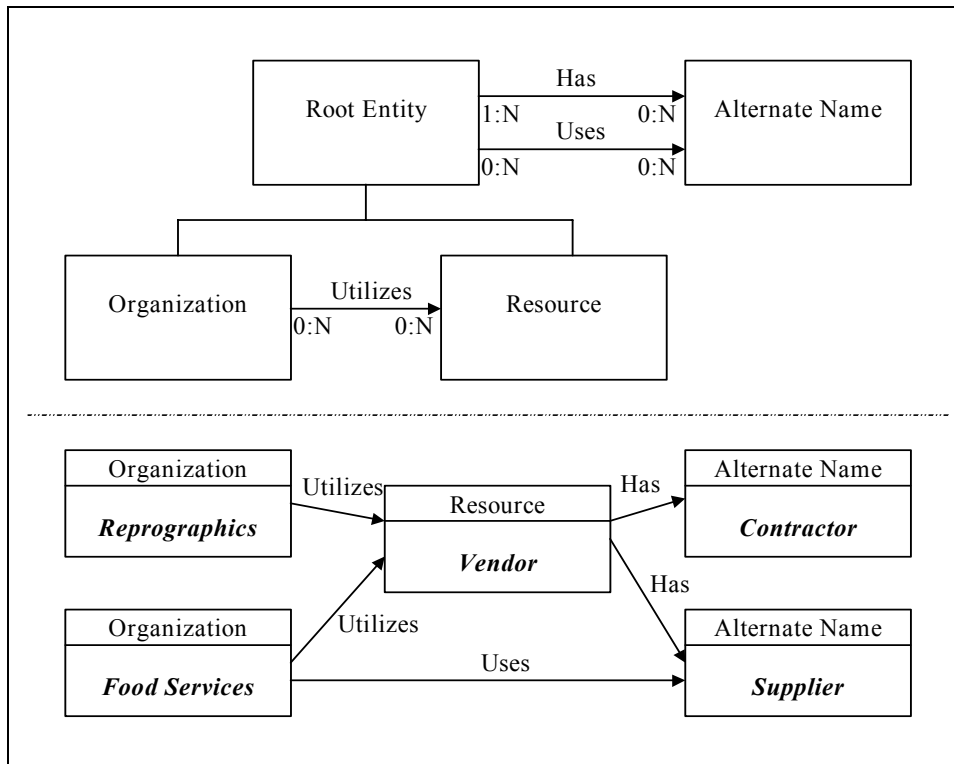


Figure 3 – AlternateName example

6.7 Audit data

The audit data supported by the Common subject area are the details of who created and updated any object. This is represented through meta-relationships from *RootEntity* to the *ToolUser* meta-entity.

6.8 Constraints

The *TextualConstraint* meta-entity enables constraints to be attached to any semantic object or set of objects. The constraint is expressed in text, either in one of a set of computable languages or in some other formal or informal language. The supported set of languages is defined in Clause 6.12 Computable languages.

6.9 Derivation

Derivations between objects can be represented through use of the *Derivation* meta-entity, using the *UsedIn* and *ProducedBy* meta-relationships to associate the objects. The *Derivation* meta-entity can be used to describe a weak form of derivation, such as instances of *Entity* being derived from instances of *Store* on a *DataFlowModel*. Stronger forms of derivation can also be described, for example where one object is an implementation of another. The derivation is expressed in text, either in one of a set of computable languages or in some other formal or informal language. These are defined in Clause 6.11, Computable Languages.

6.10 Abstraction levels

The abstraction level at which any object is used or defined in a user model can be defined through the use of the meta-entity *AbstractionLevel*. The name of the abstraction, for example **Conceptual**, **Logical** or **Physical**, is held in the *Name* meta-attribute of an instance of *AbstractionLevel*. All the meta-entities used at that level are then connected to it through the meta-relationship *SemanticInformationObject.IsCategorizedIn.AbstractionLevel*. A given meta-entity instance can be used in multiple abstraction levels; for example the same definition of Entity **Customer** may be valid at both a conceptual and logical level.