

AS/NZS ISO/IEC 15474.2:2003
ISO/IEC 15474-2:2002
Reconfirmed 2014

Australian/New Zealand Standard™

**Information technology—CDIF
framework**

Part 2: Modelling and extensibility



Standards Australia



**STANDARDS
Documents**
Pacifika Aotearoa

Document Sharing Hub

AS/NZS ISO/IEC 15474.2:2003

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 19 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 Standards may also be withdrawn.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 03133.

STANDARDS AUSTRALIA/STANDARDS NEW ZEALAND

RECONFIRMATION

OF

AS/NZS ISO/IEC 15474.2:2003
Information technology—CDIF framework
Part 2: Modelling and extensibility

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 framework

Part 2: Modelling and extensibility

First published as AS/NZS ISO/IEC 15474.2:2003.

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 5287 X

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

The objective of this Standard is to define the CDIF meta-metamodel and describe the rules and notations used throughout the CDIF family of standards. The rules for extending the CDIF semantic metamodel are also defined.

This Standard is Part 2 of AS/NZS ISO/IEC 15474, *Information technology—CDIF framework*, which is published in parts as follows:

Part 1: Overview

Part 2: Modelling and extensibility (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
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
3	Normative references.....	2
4	Terms and definitions.....	3
4.1	From other standards.....	3
4.1.1	ISO/IEC 15474-1	3
4.1.2	ISO/IEC 13238-1	3
4.2	For this standard	3
5	Symbols (and abbreviated terms).....	4
5.1	Naming and diagramming conventions	4
5.2	Abbreviations	4
6	CDIF meta-metamodel concepts and facilities.....	4
6.1	Introduction	4
6.2	Modelling approach.....	4
6.2.1	Introduction	4
6.2.2	Subject areas.....	5
6.2.3	Many-to-many meta-relationships	5
6.2.4	Attributed meta-relationships.....	5
6.2.5	Subtyping of meta-entities	5
6.2.6	Multiple inheritance in meta-entities.....	6
6.2.7	Subtyping of meta-relationships	7
6.2.8	Multiple inheritance of meta-relationships.....	8
6.2.9	Arity of meta-relationships.....	8
6.2.10	Mutual exclusivity of meta-relationships.....	8
6.3	Subject areas.....	8
6.3.1	Explicit and implicit use of <i>CollectableMetaObjects</i> in subject areas	8
6.3.2	Rules for defining subject area.....	9
6.3.3	Working metamodel	10
6.3.4	Subject area diagrams	10
6.4	Metamodel extensibility	10
6.4.1	Introduction	10
6.4.2	Extensibility	11
6.5	Exporter responsibilities.....	12
6.5.1	Introduction	12
6.5.2	Extensibility	12
6.5.3	Maximum output.....	12
6.5.4	Meta-attributes.....	12
6.6	Importer responsibilities	12
6.6.1	Introduction	12
6.6.2	Information retention.....	12
6.6.3	Working metamodel	12
6.6.4	Meta-attributes.....	13
7	Modelling conventions.....	13
7.1	Naming of meta-objects	13
7.1.1	Names	13
7.1.2	Uniqueness of names.....	13
7.2	Meta-object graphical notation	13
7.2.1	Introduction	13
7.2.2	Meta-entity symbol	14

7.2.3	Meta-relationship symbol.....	14
7.2.4	Meta-relationship cardinality symbol	15
7.2.5	Meta-entity subtype hierarchy symbol	16
7.2.6	Meta-relationship subtype hierarchy.....	16
7.2.7	Mutual exclusivity of meta-relationships	17
7.2.8	Instance diagrams.....	17
7.3	Meta-object definition format.....	19
7.3.1	Introduction	19
7.3.2	Subject area definition	19
7.3.3	Meta-entity definition.....	20
7.3.4	Meta-attribute definition.....	21
7.3.5	Meta-relationship definition	22
7.4	Meta-object reference format.....	23
7.5	Meta-object summary format	24
7.5.1	Introduction	24
7.5.2	<i>AttributableMetaObject</i> classification hierarchy.....	24
7.5.3	MetaEntity and MetaRelationship summaries.....	24
8	Meta-metamodel overview	25
8.1	Introduction	25
8.2	Meta-metamodel description	25
8.3	Meta-metamodel diagram.....	26
9	Meta-metamodel summary.....	26
9.1	<i>AttributableMetaObject</i> classification hierarchy.....	26
9.1.1	Introduction	26
9.1.2	Classification hierarchy	27
9.2	Meta-meta-entity summary	27
9.3	Meta-meta-relationship summary	29
10	Meta-metamodel specification.....	30
10.1	Data types specification.....	30
10.1.1	Introduction	30
10.1.2	Boolean.....	30
10.1.3	Date	30
10.1.4	Enumerated	30
10.1.5	Float	31
10.1.6	Identifier.....	31
10.1.7	Integer.....	31
10.1.8	Point.....	31
10.1.9	String	31
10.1.10	Text	31
10.1.11	Time	31
10.2	Meta-meta-entity and Meta-meta-attribute specifications	32
10.2.1	<i>AttributableMetaObject</i>	32
10.2.2	<i>CollectableMetaObject</i>	33
10.2.3	<i>MetaAttribute</i>	33
10.2.4	<i>MetaEntity</i>	35
10.2.5	<i>MetaObject</i>	36
10.2.6	<i>MetaRelationship</i>	39
10.2.7	<i>SubjectArea</i>	41
10.3	Meta-meta-relationship specifications	42
10.3.1	<i>AttributableMetaObject.HasSubtype.AttributableMetaObject</i>	42
10.3.2	<i>CollectableMetaObject.IsDefinedIn.SubjectArea</i>	43
10.3.3	<i>CollectableMetaObject.IsUsedIn.SubjectArea</i>	44
10.3.4	<i>MetaAttribute.IsLocalMetaAttributeOf.AttributableMetaObject</i>	45
10.3.5	<i>MetaRelationship.HasDestination.MetaEntity</i>	45
10.3.6	<i>MetaRelationship.HasSource.MetaEntity</i>	46

Table of Illustrations

Figure 1 – CDIF family of standards	1
Figure 2 – Multiple inheritance	6
Figure 3 – Multiple inheritance: Invalid meta-attribute, <i>Z1</i>	7
Figure 4 – Example of CDIF graphical notation	14
Figure 5 – The meta-entity symbol	14
Figure 6 – The meta-relationship symbol	15
Figure 7 – A Reflexive meta-relationship	15
Figure 8 – The Meta-relationship cardinality symbol	15
Figure 9 – The Meta-entity subtype hierarchy symbol	16
Figure 10 – The Meta-entity subtype hierarchy symbol: Multiple inheritance	16
Figure 11 – Mutual exclusivity of meta-relationships	17
Figure 12 – Conceptual Diagram and an Instance Diagram	18
Figure 13 – The CDIF meta-metamodel	26

Table of Tables

Table 1 – Inheritance of meta-meta-attribute values by subtype meta-entities	6
Table 2 – Inheritance of meta-meta-attribute values by subtype meta-relationships	8
Table 3 – Contents of subject area definition	19
Table 4 – Contents of meta-entity definition	20
Table 5 – Contents of meta-attribute definition	21
Table 6 – Contents of meta-relationship definition	22
Table 7 – Contents of meta-entity references	23
Table 8 – Contents of meta-relationship references	23
Table 9 – Data types supported by CDIF	30

AUSTRALIAN/NEW ZEALAND STANDARD

Information technology — CDIF framework —

Part 2: Modelling and extensibility

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 CDIF family of standards includes a semantic metamodel and a transfer format definition. It also includes the specification of a meta-metamodel and associated rules that define a framework for the semantic metamodel and the transfer format. The language that is defined for the transfer format also has applicability as a general language for Import/Export for repositories. The CDIF semantic metamodel also has applicability as the basis of standard definitions for use in repositories.

The standards that 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.

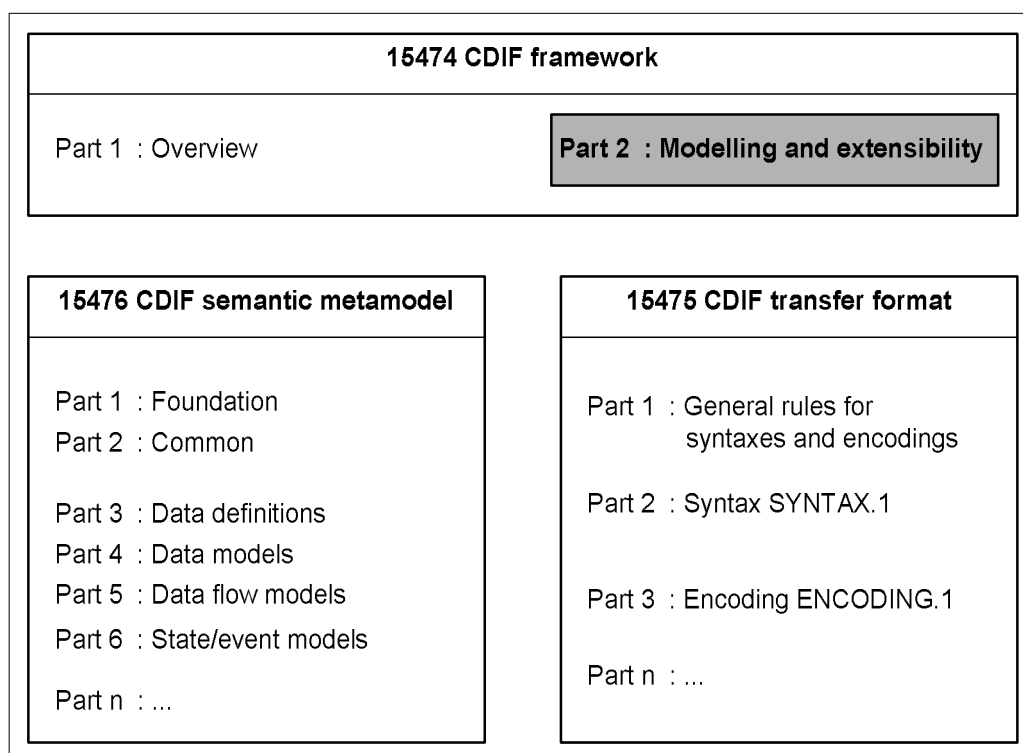


Figure 1 – CDIF family of standards

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 document (ISO/IEC 15474-2:2002, *Information technology - CDIF framework - Part 2: Modelling and extensibility*) includes the definition of the CDIF meta-metamodel and describes the rules and notations used throughout the CDIF family of standards. The rules for extending the CDIF semantic metamodel are also defined.

This document is intended to be used by anyone wishing to understand and/or use CDIF. This document provides an introduction to the entire CDIF family of standards. 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.

The document, ISO/IEC 15474-1:2002, *Information technology - CDIF framework - Part 1: Overview*, and this 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 CDIF architecture conformant if and only if it can, as a property of that product, represent the product's metamodel instances, and/or the product's metamodel using the concepts defined in the ISO/IEC 15474-2:2002, *Information technology — CDIF framework — Part 2: Modelling and extensibility* ("Framework document"), and all the concepts defined in the standard ISO/IEC 15476-1:2002, *Information technology — CDIF semantic metamodel — Part 1: Foundation* ("Foundation document"), and obeys all the constraints and rules for metamodels and meta-data defined in the Framework document, and obeys all the rules and constraints defined in the Foundation document. Conformance to the graphical notation as defined in the Framework document is not required.

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 15474. 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 15474 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 15474-1:2002, *Information technology — CDIF framework — Part 1: Overview*

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

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

¹⁾ To be published.

4 Terms and definitions

For the purposes of this part of ISO/IEC 15474, the following definitions apply. Unless otherwise noted, the definitions are specific to this part of ISO/IEC 15474.

4.1 From other standards

4.1.1 ISO/IEC 15474-1

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

Arity
 Associative entity
 Attributed relationship
 Cardinality
 CDIF
 CDIF family of standards
 CDIF graphical notation
 CDIF semantic metamodel
 CDIF meta-metamodel
 Characteristic entity
 Constraint
 Encoding
 Entity
 Instance
 Semantic metamodel
 Kernel entity
 Meta-attribute
 Meta-entity
 Meta-meta-attribute
 Meta-meta-entity
 Meta-metamodel
 Meta-meta-relationship
 Metamodel
 Meta-object
 Meta-relationship
 Model
 Model layers
 N-ary relationship
 Relationship
 Role
 Subject area
 Syntax
 Working metamodel

4.1.2 ISO/IEC 13238-1

This part of ISO/IEC 15474 makes use of the following terms from ISO/IEC 13238-1:

Transfer file
 CDIF transfer file
 Exporter
 Importer

4.2 For this standard

For the purpose of this part of ISO/IEC 15474 new terms are defined when introduced. Double quotes are used to introduce new terms (e.g., "model layer").

5 Symbols (and abbreviated terms)

5.1 Naming and diagramming conventions

Conventions for naming, diagramming, describing and defining meta-objects can be found in Clause 7 of this document. Unless otherwise noted, they are applicable to meta-meta-objects named, diagrammed, described and defined in this document.

5.2 Abbreviations

The following abbreviations are used in this international standard:

CDIF CASE Data Interchange Format (originally)

ERA Entity-Relationship-Attribute modelling

6 CDIF meta-metamodel concepts and facilities

6.1 Introduction

This clause describes the concepts of the CDIF meta-metamodel. An overview and a diagram of the CDIF meta-metamodel are included in Clause 8 Meta-metamodel overview. A summary of the meta-metamodel is included in Clause 9 Meta-metamodel summary. The formal definition can be found in Clause 10 Meta-metamodel specification.

The CDIF meta-metamodel is defined using the same facilities that are used to define the CDIF metamodel (and could be used to define models). It is, therefore, quite common to get the names of the objects at the different meta-layers confused. To minimize this problem, the reader should remember the following:

- The components of the meta-metamodel are known as meta-meta-objects (i.e., meta-meta-entities, meta-meta-relationships and meta-meta-attributes);
- The components of the metamodel are known as meta-objects (i.e., meta-entities, meta-relationships and meta-attributes);
- The components of models are known as objects (e.g., processes, entities, flows);
- The names of the components in the meta-metamodel usually have a single meta- as a prefix (e.g., there is a meta-meta-entity, in the meta-metamodel, known as *MetaAttribute*);
- The names of the components in the metamodel usually have no meta prefix.

The CDIF meta-metamodel is the definition of the concepts used to develop CDIF-conformant metamodels. This includes the Entity-Relationship-Attribute diagramming technique and information definition requirements pertinent to the metamodel development process.

6.2 Modelling approach

6.2.1 Introduction

Because the CDIF meta-metamodel is used to define metamodels, it must be capable of expressing a complete description of a metamodel. A CDIF conformant metamodel contains three types of objects: entities, relationships and attributes; these objects in the metamodel are properly known as meta-entities, meta-relationships and meta-attributes. These objects are used to model the information content of the metamodel, and in fact, it is a rule in CDIF, that any single concept, or set of information may only be represented one way; that is, although there may be several ways of modelling a concept, only one will be included in the model.

The meta-metamodel is composed of two things:

- An Entity-Relationship-Attribute model that is used to describe the CDIF metamodel. This Entity-Relationship-Attribute model is also self-describing (i.e., is defined using itself).
- A graphical notation for representing occurrences (i.e., instances) of the meta-metamodel. This notation is used in CDIF in two ways: to depict the contents of subject areas in the CDIF metamodel, and to depict the CDIF meta-metamodel itself.

It is important for the reader to remember that the rules in this clause apply only to the metamodels defined using the CDIF meta-metamodel. In particular, these rules do not relate to Entity-Relationship-Attribute models that are transferred during a CDIF transfer (see ISO/IEC 15476-4:—, *Information technology — CDIF semantic metamodel — Part 4: Data models* for further information).

This clause describes some general modelling rules that are part of the definition of the CDIF meta-metamodel. They are described with a meta- prefix. The rules apply to the CDIF semantic metamodel and all extensions defined using the extensibility mechanism. The rules are also adhered to by the CDIF meta-metamodel²⁾.

6.2.2 Subject areas

The CDIF semantic metamodel is a single integrated model that is a description of the set of concepts and notations used to define an information model. This model is huge; therefore, it is partitioned into manageable sections by building views of the underlying metamodel. These partitions, or "subject areas" as they are known, group together collectable meta-objects (meta-entities, meta-relationships and meta-attributes) that address a particular aspect within the Systems Development Life Cycle (e.g., Data Flow Modelling), or that provide similar functionality (e.g., Data Definition). Meta-objects may be shared by many subject areas. They are defined in one subject area, and referred to in other subject areas.

6.2.3 Many-to-many meta-relationships

The CDIF meta-metamodel supports many-to-many meta-relationships.

6.2.4 Attributed meta-relationships

The CDIF meta-metamodel supports attributed meta-relationships.

6.2.5 Subtyping of meta-entities

The CDIF meta-metamodel provides support for subtyping (sometimes known as specialization) and its converse, supertyping (sometimes known as generalization).

In particular, a meta-entity may be defined to be a "subtype" of another meta-entity. The subtype meta-entity inherits all of the meta-attributes and meta-relationships of its immediate and indirect supertype meta-entities.

Supertyping is used as a classification mechanism to group meta-objects with a substantial level of common semantics; an importer can use this classification as an aid to deciding how to represent any meta-object that is not directly representable within the importing tool.

Subtype meta-entities have unique *CDIFMetaIdentifiers*; that is, the *CDIFMetaIdentifier* is not inherited from the parent. The *Description*, *Constraints*, *Usage*, and *Aliases* that were defined for the parent meta-entity are inherited. These textual fields may also be expanded in the local definition of the subtype meta-entity. Subtype meta-entities inherit all the meta-attributes that have been defined for the parent meta-entity, and all those of its antecedents. The characteristics of the meta-attribute (for example, *Name*) are unchanged when inherited by a subtype meta-entity. Additional, local, meta-attributes may be defined specific to the subtype meta-entity.

²⁾ The CDIF modelling rules may be used at any meta-level; they may be used to create models.

In the same way that meta-attributes are inherited, any meta-relationships that are defined with the parent meta-entity as the source or destination are inherited by subtype meta-entities. None of the characteristics of the meta-relationship are changed by its use by a subtype; that is, the *Name*, *Cardinality*, *CDIFMetaIdentifier*, *Aliases*, *Constraints*, *Description*, and *Usage*, are unchanged. Notice that even the full name of the meta-relationship is unchanged and will contain the name of the supertype meta-entity rather than that of the subtype.

A summary of the inheritance rules for subtype meta-entities is shown in Table 1.

Table 1 – Inheritance of meta-meta-attribute values by subtype meta-entities

META-META-ATTRIBUTE	Inherited by Subtype Meta-entity
ALIASES	Yes, may be extended
CDIFMETAIDENTIFIER	No
CONSTRAINTS	Yes, may be extended
DESCRIPTION	Yes, may be extended
ISABSTRACT	No
NAME	No
TYPE	No
USAGE	Yes, may be extended

6.2.6 Multiple inheritance in meta-entities

A meta-entity may have multiple supertypes, allowing for "multiple inheritance". In this case, the subtype inherits all of the meta-attributes and meta-relationships of all of its supertypes (and their supertypes, etc.).

There are several situations that occur when using multiple-inheritance that either seem to, or do, present inheritance conflicts.

When several inheritance threads of a meta-entity converge at a common supertype, the inheriting meta-entity does not inherit multiple copies of the meta-attributes and meta-relationships from that point to the top of the hierarchy; it only inherits one copy. Figure 2 shows meta-entity **A** having two subtype meta-entities (**B** and **C**), which are both supertype meta-entities of meta-entity **D**. Both local and inherited meta-attributes for all meta-entities are shown.

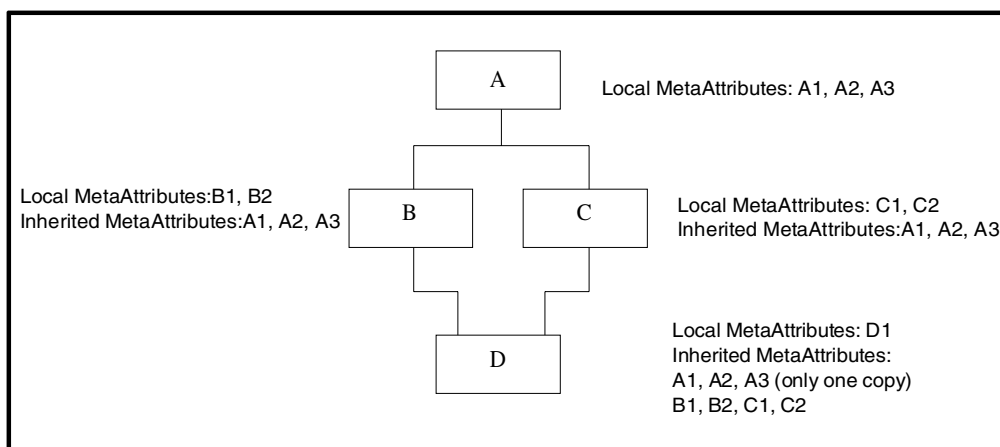


Figure 2 – Multiple inheritance

The other potential conflict occurs when a meta-entity inherits meta-attributes from two supertypes, both having a meta-attribute with the same name that are, in fact, different meta-attributes. The behaviour for CDIF is undefined in these circumstances. To guarantee proper (predictable) behaviour from importers and exporters, this situation

must be avoided. The CDIF semantic metamodel and CDIF meta-metamodel are defined ensuring that this does not occur.

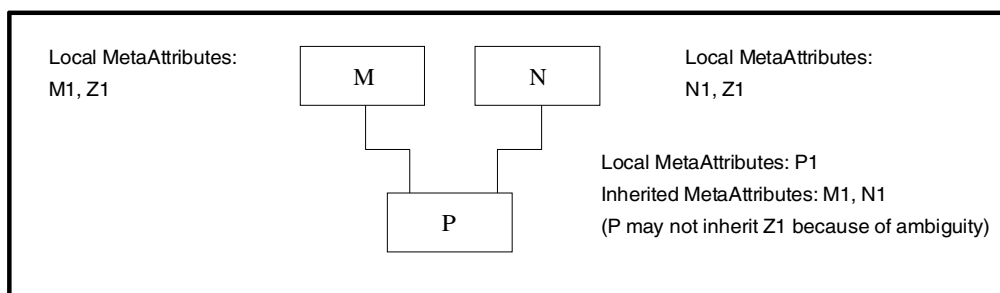


Figure 3 – Multiple inheritance: Invalid meta-attribute, Z1

Figure 3 shows an invalid inheritance example, where the meta-entity **P** has two supertype meta-entities (**M** and **N**) that both have an identically named meta-attribute, **Z1**.

6.2.7 Subtyping of meta-relationships

A meta-relationship may be defined to be a subtype of another meta-relationship.

A subtype meta-relationship may be defined at the 'level' in the meta-entity hierarchy where the supertype meta-relationship was defined (that is, with the same source and destination meta-entities as the supertype). Alternatively, the source meta-entity or destination meta-entity, or both, may be subtypes of the source and destination meta-entities in the supertype meta-relationship.

Each subtype meta-relationship has a unique *CDIFMetaIdentifier*; that is, it does not inherit the *CDIFMetaIdentifier* from the supertype meta-relationship.

The cardinalities of the subtype meta-relationship are defined locally; however, they may only be as constrained, or more constrained than at the supertype level. Therefore, the minimum cardinality of the subtype must be greater than, or equal to, that of the supertype, and the maximum cardinality must be less than, or equal to, that of the supertype.

The other, textual, characteristics of the subtype meta-relationship are inherited from the supertype meta-relationship but may be further refined for the subtype.

The immediate and indirect subtype meta-relationship inherits any meta-attributes that have been defined for the supertype meta-relationship. None of the characteristics of the meta-attributes are changed. Additional meta-attributes may be created for the subtype meta-relationship.

A summary of the inheritance rules for subtype meta-relationships is shown in Table 2.

Table 2 – Inheritance of meta-meta-attribute values by subtype meta-relationships

META-META-ATTRIBUTE	Inherited by Subtype Meta-relationship
ALIASES	Yes, may be extended
CDIFMETAIDENTIFIER	No
CONSTRAINTS	Yes, may be extended
DESCRIPTION	Yes, may be extended
ISABSTRACT	No
MAXDESTCARD	No, must be locally defined
MAXSOURCECARD	No, must be locally defined
MINDESTCARD	No, must be locally defined
MINSOURCECARD	No, must be locally defined
NAME	No
USAGE	Yes, may be extended

Between any pair of meta-entities, there may be both a supertype meta-relationship (possibly due to inheritance by a subtype meta-entity) and a subtype meta-relationship defined. There may not be an instance of both the supertype meta-relationship and the subtype meta-relationship between the same pair of meta-entity instances. However, in most cases it is expected that the supertype meta-relationship will be abstract (not instantiable), which means that the condition of an instance of both supertype and subtype meta-relationships between the same instance pair will not arise.

6.2.8 Multiple inheritance of meta-relationships

Meta-relationships may inherit from multiple supertypes in the same way that meta-entities may inherit from multiple supertypes. The discussion in Clause 6.2.6, Multiple inheritance in meta-entities, that describes potential inheritance conflicts arising from multiple inheritance applies equally to multiple inheritance of meta-relationships, and the diagrams, although depicting meta-entities (in boxes) may be read as though the boxes represent meta-relationships.

6.2.9 Arity of meta-relationships³⁾

The CDIF semantic metamodel, and any extensions, must be defined using binary meta-relationships rather than n-ary meta-relationships. (Note that the functionality of n-ary relationships is covered in the Data Models subject area in the CDIF semantic metamodel, see ISO/IEC 15476-4:—, *Information technology — CDIF semantic metamodel — Part 4: Data models*.)

6.2.10 Mutual exclusivity of meta-relationships

In metamodels mutual exclusivity of meta-relationships is documented as a constraint on the meta-relationships that are in the exclusivity group.

6.3 Subject areas

6.3.1 Explicit and implicit use of *CollectableMetaObjects* in subject areas

Subject areas may contain some *CollectableMetaObjects* that are explicitly defined and/or used and, may contain some that are implicitly used through inheritance.

A meta-object is used in the subject area (and thus included in Clause 10, Specification of the subject area) when:

³⁾ The arity of a relationship may also be called the degree of the relationship. A binary relationship has an arity of two; a ternary relationship has an arity of three, etc.

- It is defined in this subject area, or
- It has a local meta-attribute defined in this subject area, or
- It has a local subtype in this subject area (these are both supertype meta-entities whose meta-attributes and meta-relationships are inherited by meta-entities that are used, and supertype meta-relationships for those meta-relationships that are used), or
- It has a local meta-relationship in this subject area.

In the standardized CDIF subject areas, which are published as standards, the explicitly used *CollectableMetaObjects* are defined or referenced in Clause 10, Subject area specification, of each subject area standard. *CollectableMetaObjects* defined in the subject area are defined in Clause 10. *CollectableMetaObjects* used, but not defined in this subject area are referenced in clause 10.

When a new subject area is being created using extensibility, instances of the *CollectableMetaObject.IsDefinedIn.SubjectArea* and *CollectableMetaObject.IsUsedIn.SubjectArea* meta-meta-relationships must be created for all explicitly used *CollectableMetaObjects*. Instances of meta-meta-relationships should not be created for implicitly used *CollectableMetaObjects* since inheritance of meta-attributes and meta-relationships is assumed all the way up the inheritance hierarchies to *RootObject*.

6.3.2 Rules for defining subject area

Subject area views are built by instantiating the *CollectableMetaObject.IsDefinedIn.SubjectArea* and *CollectableMetaObject.IsUsedIn.SubjectArea* meta-meta-relationships for any *MetaEntity*, *MetaRelationship*, and *MetaAttribute* that are explicitly used (defined or referred to). The following rules should be observed when defining the scope of a subject area:

- Where a meta-relationship is explicitly used in a subject area, both the source and destination meta-entities must also be used.
- Where both source and destination meta-entities for a meta-relationship happen to be explicitly used in a subject area, there is no obligation to use the meta-relationship.
- Where a meta-entity is explicitly used in a subject area and it participates in another semantic metamodel in a mandatory meta-relationship (at the mandatory end), the meta-relationship need not be used in the subject area.
- Where a meta-attribute is explicitly used in a subject area, the meta-entity or meta-relationship that the meta-attribute describes must also be explicitly used in the subject area.
- Where a meta-entity or meta-relationship is explicitly used in a subject area, any local meta-attributes that describe it that are required by the subject area must be explicitly used.
- Where a meta-entity or meta-relationship is explicitly used in a subject area, there is no obligation to use any of its meta-attributes, even mandatory ones.
- All inherited meta-attributes for both meta-entities and meta-relationships will be implicitly used in the subject area all the way up the inheritance hierarchy. The *CollectableMetaObject.IsDefinedIn.SubjectArea* and *CollectableMetaObject.IsUsedIn.SubjectArea* meta-meta-relationship should not be instantiated for inherited meta-attributes.
- For subtype meta-entities, any meta-relationships in which the supertype meta-entity participates will be implicitly used in a subject area, provided the other meta-entity, or a subtype of it, is explicitly used.

Note that where two subject areas are used in a transfer, a meta-relationship between two implicitly used supertype meta-entities (one in each subject area) will be inherited between the appropriate subtype meta-entities in the working metamodel for the transfer.