

Information technology — Specification for standardization and registration of data elements and associated metadata — Part 3: Basic attributes and registry metamodel

Technologie de l'information — Spécifications pour la normalisation et l'enregistrement d'éléments de données et de leurs métadonnées — Partie 3 : Attributs de base et métamodèle de registre

Document type: International Standard
Document subtype:
Document stage: (50) Approval
Document language: E

Copyright notice

This ISO document is a Draft International Standard and is copyright-protected by ISO. Except as permitted under the applicable laws of the user's country, neither this ISO draft nor any extract from it may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, photocopying, recording or otherwise, without prior written permission being secured.

Requests for permission to reproduce should be addressed to ISO at the address below or ISO's member body in the country of the requester.

*Copyright Manager
ISO Central Secretariat
1 rue de Varembé
1211 Geneva 20 Switzerland
tel. + 41 22 749 0111
fax + 41 22 734 1079
internet: iso@iso.ch*

Reproduction may be subject to royalty payments or a licensing agreement.

Violators may be prosecuted.

Contents

Foreword	vi
Introduction.....	vii
1 Scope	1
1.1 Scope – Structure of a metadata registry.....	1
1.2 Scope – Basic attributes of metadata items.....	1
1.3 Scope – Exclusions.....	2
1.4 Areas of Applicability	2
2 Normative references	2
3 Definitions.....	4
3.1 Definitions of Metamodel Constructs.....	4
3.2 Broad Terms used in this part of ISO/IEC 11179.....	5
3.3 Alphabetical list of metadata objects in the metamodel.....	8
3.4 List of Abbreviations	27
4 Structure of a Metadata Registry	27
4.1 Metamodel for a Metadata Registry	27
4.2 Application of the metamodel	28
4.3 Specification of the metamodel	28
4.4 Types and Instances	28
4.5 Extensibility	29
4.6 Date References	29
4.7 Description of metamodel.....	29
4.8 Administration and Identification region.....	32
4.9 Naming and Definition Region	38
4.10 Classification Region	41
4.11 Data Element Concept Region.....	43
4.12 Conceptual and Value Domain Region.....	46
4.13 Data Element Region.....	51
4.14 Metamodel Overview	55
5 Basic Attributes of Metadata.....	56
5.1 Use of basic attributes	56
5.2 Common attributes.....	56
5.3 Attributes specific to Data Element Concepts	58
5.4 Attributes specific to Data Elements	59
5.5 Attributes specific to Conceptual Domains.....	59
5.6 Attributes specific to Value Domains	59
5.7 Attributes specific to Permissible Values	59
5.8 Attributes specific to Value Meanings.....	60
6 Conformance	61
6.1 Degree of Conformance	61
6.2 Levels of Conformance	62
6.3 Obligation	62
6.4 Conformance to prior editions of this Standard	62
6.5 Implementation Conformance Statement (ICS).....	63
6.6 Roles and Responsibilities for Registration.....	63
Annex A (Informative) Alphabetical List of Terms.....	65
Annex B (Informative) Modelling Notation.....	69

B.1	Modelling symbols	69
Annex C (Informative)	Mapping the ISO/IEC 11179-3:1994 basic attributes to the ISO/IEC 11179-3:2002 metamodel and basic attributes	75
C.1	Introduction	75
C.2	Mapping the Basic Attributes.....	78
Bibliography.....		107

Table of Figures

Figure 1: Common facilities for all Administered Items.....	30
Figure 2: Types of Administered Items	31
Figure 3: Administration and identification metamodel region.....	32
Figure 4: Administration and identification region – Encapsulated classes used as datatypes	33
Figure 5: Naming and Definition metamodel region	38
Figure 6: Classification metamodel region	41
Figure 7: Data Element Concept metamodel region.....	43
Figure 8: Conceptual and value domain metamodel region.....	47
Figure 9: Data Element metamodel region.....	52
Figure 10: High-level metamodel.....	55
Figure B-1: Sample modelling diagram.....	69
Figure B-2: Notation for "Class"	70
Figure B-3: Notation for "Association"	70
Figure B-4: Notation for relationship between Classes.....	70
Figure B-5: Notation for relationship with Cardinality.....	70
Figure B-6: Notation for "Association Class"	71
Figure B-7: Notation for Supertype / Subtypes.....	71
Figure B-8: Notation for Aggregation	72
Figure B-9: Notation for Composite Aggregation.....	72
Figure B-10: Notation for Class with Attributes.....	73
Figure B-11: Notation for Attribute Capsules.....	73
Figure C-1: Basic Attributes of Data elements	75

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. 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/IEC 11179 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 11179-3 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 32, *Data Management and Interchange*.

This second edition cancels and replaces the first edition (ISO/IEC 11179-3:1994), clauses, subclauses, tables, figures, and annexes of which have been technically revised.

ISO/IEC 11179 consists of the following parts, under the general title *Information technology — Specification for standardization and registration of data elements and associated metadata*:

- Part 1: Framework for the specification and standardization of data elements
- Part 2: Classification for data elements
- *Part 3: Basic attributes and registry metamodel*
- Part 4: Rules and guidelines for the formulation of data definitions
- Part 5: Naming and identification principles for data elements
- Part 6: Registration of data elements

NOTE The other parts of ISO/IEC 11179 are undergoing revision and their titles may change in the process.

There are three Informative Annexes to this part of ISO/IEC 11179:

- Annex A – Alphabetical List of Terms
- Annex B – Modeling Notation
- Annex C – Mapping the 11179-3:1994 basic attributes to the 11179-3:2002 metamodel and basic attributes.

Introduction

Data processing and electronic data interchange rely heavily on accurate, reliable, controllable and verifiable data recorded in databases. A prerequisite for correct and proper use and interpretation of data is that both users and owners of data have a common understanding of the meaning and representation of the data. To facilitate this common understanding, a number of characteristics, or attributes, of the data have to be defined. These characteristics of data are known as “metadata”, that is, “data that describes data”. To facilitate such common understandings this part of ISO/IEC 11179 provides for the attributes of data elements and associated metadata to be specified and registered as metadata items in a metadata registry.

This part of ISO/IEC 11179 specifies the structure of a metadata registry in the form of a conceptual data model. The metadata registry is used to keep information about data elements and associated concepts, such as “data element concepts”, “conceptual domains” and “value domains”. Generically, these are all referred to as “metadata items”. Such metadata are necessary to clearly describe, record, analyse, classify and administer data.

When considering data and metadata, it is important to distinguish between types of data/metadata, and instances of these types. Clause 4 of this part of ISO/IEC 11179 specifies the types of metadata objects that form the structure of a metadata registry. A metadata registry will be populated with instances of these metadata objects (metadata items), which in turn define types of data, e.g. in an application database. In other words, instances of metadata specify types of application level data. In turn, the application database will be populated by the real world data as instances of those defined data types.

NOTE ISO/IEC 10027:1990 IRDS Framework explains the concepts of different levels of modelling.

This part of ISO/IEC 11179 also describes the basic attributes of metadata items for purposes where a complete registry is not appropriate.

This part of ISO/IEC 11179 is of interest to information developers, information managers, data administrators, standards developers and others who are responsible for making data understandable and shareable. ISO/IEC 11179 has broad applicability across subject area domains and information technologies.

Information technology — Specification for standardization and registration of data elements and associated metadata — Part 3: Basic attributes and registry metamodel

1 Scope

Sub-clauses 1.1 and 1.2 outline the major topics within the scope of this part of ISO/IEC 11179.

Sub-clause 1.3 identifies aspects not currently addressed.

Sub-clause 1.4 provides examples of activities where this part of ISO/IEC 11179 may be applied.

1.1 Scope – Structure of a metadata registry

The primary purpose of this part of ISO/IEC 11179 is to specify the structure of a metadata registry. A comprehensive metadata registry management function requires a set of rules and procedures. These rules and procedures are set out in the following Clauses and Annexes and are complemented elsewhere in this document as follows:

- a) the definitions of metadata objects are in Clause 3.3 of this part of ISO/IEC 11179;
- b) the structure of the registry in the form of a conceptual data model is in Clause 4 of this part of ISO/IEC 11179;

Aspects of the registry are expanded on in other parts of ISO/IEC 11179, as follows:

- a) rules and guidelines for classifying metadata are in ISO/IEC 11179-2;
- b) rules and guidelines for the formulation of definitions are in ISO/IEC 11179-4;
- c) naming and identifying principles for metadata are in ISO/IEC 11179-5;
- d) rules and guidelines for registering metadata are in ISO/IEC 11179-6.

While the model diagrams are presented in UML format, this part of ISO/IEC 11179 does not assume nor endorse any specific system environment, database management system, database design paradigm, system development methodology, data definition language, command language, system interface, user interface, computing platform, or any technology required for implementation. This part of ISO/IEC 11179 does not directly apply to the actual use of data in communications and information processing systems.

1.2 Scope – Basic attributes of metadata items

This part of ISO/IEC 11179 also specifies basic attributes which are required to describe metadata items, and which may be used in situations where a complete metadata registry is not appropriate (e.g. in the specification of other International Standards). These basic attributes are described in clause 5.

1.3 Scope – Exclusions

This part of ISO/IEC 11179 does not currently support the following requirements.

- a) Complex data structures, encapsulation, stereotyping and inheritance;
- b) Ability to enforce uniqueness of names within a Context;
- c) Specification of Naming Conventions for a Context;
- d) Specification of Time in addition to Date;
- e) Prescribed conceptual domains and value domains for the attributes in the metamodel;
- f) XML documents
- g) Application Programming Interfaces (APIs) and associated bindings to access a registry.

It is anticipated that some or all of these requirements will be addressed in future editions of this part of ISO/IEC 11179, or in companion standards or technical reports.

1.4 Areas of Applicability

This part of ISO/IEC 11179 applies to activities including:

- a) the definition, specification and contents of metadata registries, including interchanging or referencing among various collections of data elements;
- b) the design and specification of application-oriented data models, databases and message types for data interchange;
- c) the actual use of data in communications and information processing systems;
- d) interchange or reference among various collections of metadata.

2 Normative references

The following standards contain provisions, which, through reference in the text, constitute provisions for this part of ISO/IEC 11179. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 11179 are encouraged to investigate the possibility of applying the most recent editions of standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

[ISO 31-0:1982](#), Quantities and Units – Part 0: General Principles

[ISO 639-2:1998](#), Codes for the representation of the names of languages – Part 2: Alpha-3 code

[ISO 1087-1:2000](#), Terminology work – Vocabulary – Part 1: Theory and Application

[ISO/IEC 2382-4:1999](#), Information technology – Vocabulary – Part 4 Organization of Data

[ISO/IEC 2382-17:1993](#), Information technology – Vocabulary Part 17: Databases

[ISO 3166-1:1997](#), Codes for the representation of names of countries and their subdivisions – Part 1: Country codes

ISO 5127:2001, Information and Documentation – Vocabulary

ISO/IEC 6523-1:1998, Information technology – Structure for the identification of organization and organization parts – Part 1: Identification of organization identification schemes.

ISO/IEC 6523-2:1998, Information technology – Structure for the identification of organization and organization parts – Part 2: Registration of organization identification schemes.

ISO 8601:2000, Data elements and interchange formats – Information exchange – Representation of dates and times

ISO/IEC 11179-1:1999, Information technology – Specification and standardization of data elements Part 1 : Framework for the specification and standardization of data elements

ISO/IEC 11179-2:1999, Information technology – Specification and standardization of data elements Part 2 : Classification of data elements

ISO/IEC 11179-4:1995, Information technology – Specification and Standardization of data elements Part 4 : Rules and guidelines for the formulation of data definitions

ISO/IEC 11179-5:1995, Information technology – Specification and standardization of data elements Part 5 : Naming and identification principles for data elements

ISO/IEC 11179-6:1997, Information technology – Specification and standardization of data elements Part 6 : Registration of data elements

ISO/IEC 11404:1996, Information technology – Language-Independent Datatypes

ISO 12620:1999, Computer applications in Terminology – Data categories

ISO 19108:2002, Geographic Information – Temporal Schema

ISO/IEC 19501-1:2001, Information technology – Unified Modeling Language (UML) – Part 1: Specification

3 Definitions

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

Sub-clause 3.1 defines metamodel constructs, used in specifying the registry metamodel.

Sub-clause 3.2 lists broader terms, and their definitions, used in this document that are not included in either sub-clause 3.1 or sub-clause 3.3.

Sub-clause 3.3 defines metadata objects prescribed by the metamodel itself.

An alphabetical list of terms from all three clauses is provided in Annex A.

3.1 Definitions of Metamodel Constructs

This sub-clause defines the metamodel constructs used in specifying the registry metamodel in clause 4.

3.1.1

association

a semantic **relationship** between two **classes**

NOTE An **association** is a type of **relationship**.

[Adapted from ISO/IEC 19501-1:2001 2.5.2.3]

3.1.2

association class

an **association** that is also a **class**

NOTE It not only connects a set of **classes**, but also defines a set of features that belong to the **relationship** itself.

[Adapted from ISO/IEC 19501-1:2001 2.5.2.4]

3.1.3

attribute

a **characteristic** of an **object** or **entity**

3.1.4

attribute capsule

an **attribute** that encapsulates other **classes** and associated **relationships**

3.1.5

class

a description of a set of **objects** that share the same **attributes**, operations, methods, **relationships**, and semantics

[ISO/IEC 19501-1:2001 2.5.2.9]

3.1.6

encapsulated class

a **class** encapsulated by an **attribute capsule**

NOTE An **encapsulated class** is used as a datatype for an **attribute capsule**.

3.1.7

generalization

a **relationship** between a more general **class** (the parent) and a more specific **class** (the child) that is fully consistent with the first **class** and that adds additional information.

NOTE A **generalization** is a type of **relationship**.

[Adapted from ISO/IEC 19501-1:2001 2.5.2.24]

3.1.8

identifier (in metadata registry)

an unambiguous, linguistically neutral sequence of characters, capable of uniquely identifying that with which it is associated, within a specified context

NOTE 1 A name should not be used as an identifier because it is not linguistically neutral.

NOTE 2 Once assigned, an identifier should not be reassigned, to avoid possible later confusion over what is being identified.

3.1.9

relationship (in registry metamodel)

a connection among model elements

NOTE In this part of ISO/IEC 11179, a relationship is either an association or a generalization.

[ISO/IEC 19501-1:2001 2.5.2.36]

3.2 Broad Terms used in this part of ISO/IEC 11179

3.2.1

attribute value

the value associated with a specific instance of an **attribute**

NOTE Amended from ISO 2382-17:1993 (17.02.13) to distinguish an instance of an attribute from its value.

3.2.2

basic attribute

an **attribute** of a **metadata item** frequently needed in its specification

3.2.3

binding

an instantiation of a mapping from one framework or specification to another

3.2.4

characteristic

abstraction of a property of an **object** or of a set of objects

NOTE Characteristics are used for describing **concepts**.

[ISO 1087-1:2000 (3.2.4)]

3.2.5

common attribute

a **basic attribute** that is applicable to all types of metadata item

3.2.6

conceptual data model

a **data model** that represents an abstract view of the real world

3.2.7

conditional

required under certain specified conditions

NOTE 1 One of three obligation statuses applied to the attributes of metadata items, indicating the conditions under which the attribute is required. See also **3.2.15 mandatory** and **3.2.26 optional**.

NOTE 2 Obligation statuses are to be enforced for metadata items with a Registration Status of "recorded" or higher.

3.2.8

data

a re-interpretable representation of information in a formalized manner suitable for communication, interpretation or processing

NOTE Data can be processed by human or automatic means.

[ISO/IEC 2382-1:1998 (01.01.02)]

3.2.9

data model

a graphical and/or lexical representation of data, specifying their properties, structure and inter-relationships

3.2.10

designation

representation of a concept by a sign which denotes it

NOTE See also **3.3.68 Designation** (of Administered Item).

[ISO 1087-1:2000 (3.4.1)]

3.2.11

entity

any concrete or abstract thing that exists, did exist, or might exist, including associations among these things

EXAMPLE A person, object, event, idea, process, etc...

NOTE Please observe that an entity exists whether data available about it are available or not.

[ISO/IEC 2382-17:1999 (17.02.05)].

3.2.12

extension

a feature not defined by this part of ISO/IEC 11179 that is added by an implementation

3.2.13

extension (of registry metamodel)

a **class**, an **attribute** or a **relationship** that an implementation of a **metadata registry** provides that is not defined by this part of ISO/IEC 11179

3.2.14

language

system of signs for communication, usually consisting of a vocabulary and rules

[ISO 5127:2001 (1.1.2.01)]

3.2.15

mandatory

always required

NOTE 1 One of three obligation statuses applied to the attributes of metadata items, indicating the conditions under which the attribute is required. See also **3.2.7 conditional** and **3.2.26 optional**.

NOTE 2 Obligation statuses are to be enforced for metadata items with a Registration Status of "recorded" or higher.

3.2.16

metadata

data that defines and describes other **data**

3.2.17

metadata item

an instance of a **metadata object**

NOTE 1 In all parts of ISO/IEC 11179, this term is applied only to instances of metadata objects described by the metamodel in clause 4 of this document. Examples include instances of Data Elements, Data Element Concepts, Permissible Values etc.

NOTE 2 A metadata item has associated attributes, as appropriate for the metadata object it instantiates.

3.2.18

metadata object

an object type defined by a metamodel

NOTE In all parts of ISO/IEC 11179, this term is applied only to metadata objects described by the metamodel in clause 4 of this document. Examples include Data Elements, Data Element Concepts, Permissible Values etc. See sub-clause 3.3 for a complete list.

3.2.19

metadata register

the information store or database maintained by a **Metadata Registry**

3.2.20

Metadata Registry (Mdr)

an information system for registering **metadata**

NOTE The associated information store or database is known as a **metadata register**.

3.2.21

metadata set

any collection of **metadata**

3.2.22

metamodel

a **data model** that specifies one or more other data models

3.2.23

metamodel construct

a modelling construct used in defining a **metamodel**

NOTE The modelling constructs used in this document are defined in sub-clause 3.1.

3.2.24

name

the **designation** of an object by a linguistic expression

3.2.25

object

anything perceivable or conceivable

NOTE Objects may also be material (e.g. engine, a sheet of paper, a diamond), immaterial (e.g. conversion ratio, a project plan) or imagined (e.g. a unicorn).

[ISO 1087-1:2000 (3.1.1)]

3.2.26

optional

allowed but not required

NOTE 1 One of three obligation statuses applied to the attributes of metadata items, indicating the conditions under which the attribute is required. See also **3.2.6 conditional** and **3.2.14 mandatory**.

NOTE 2 Obligation statuses are to be enforced for metadata items with a Registration Status of "recorded" or higher.

3.2.27

registry item

a **metadata item** recorded in a **metadata registry**

3.2.28

registry metamodel

a **metamodel** specifying a **metadata registry**

3.2.29

related metadata reference

a **reference** from one **metadata item** to another

NOTE The metamodel allows any **Administered Item** to reference any other **Administered Item** in the same **Metadata Registry** via the **Administered Item Relationship**. For an external reference, a **Registration Authority** could choose to use a **Reference Document**, an **administrative note** or an **explanatory comment**.

3.2.30

stewardship (of metadata)

the responsibility for the determination of the content of **Administration Records** applicable to one or more **Administered Items**

NOTE 1 The responsibility for the registration of metadata may be different from the responsibility for stewardship of metadata.

NOTE 2 See also **3.3.126 Stewardship** (of Administered Item).

3.3 Alphabetical list of metadata objects in the metamodel

This sub-clause provides definitions for terms which are the names of metadata objects in the metadata model in clause 4. Each metadata object is modelled on one of the metamodel constructs from sub-clause 3.1 (i.e. classes, attributes, relationships, attribute capsules or association classes). The metamodel construct applicable to each metadata object is indicated after the definition. For attributes, the associated class is also identified.

This sub-clause follows the capitalization convention of the model, which is to capitalize the names of classes and association classes, but not attributes or relationships.

3.3.1

Administered Item

a **registry item** for which administrative information is recorded in an **Administration Record**

NOTE 1 Metamodel construct is: *Class*.

NOTE 2 The types of *Administered Item* specified by this part of ISO/IEC 11179 are listed in clause 4.6.2.

3.3.2

administered item classification

the **relationship** where an **Administered Item** is classified based on a specified **Classification Scheme**

NOTE Metamodel construct is: *Relationship*.

3.3.3

administered item context

the **relationship** that provides a **Context** for an **Administered Item**

NOTE Metamodel construct is: *Relationship*.

3.3.4

administered item identifier

an **identifier** for an **administered item**

NOTE Metamodel construct is: *Attribute of Administration Record*.

3.3.5

Administered Item Relationship

the **relationship** among two or more **Administered Items**

NOTE Metamodel construct is: *Association Class*.

3.3.6

administered item relationship type description

the description of the type of **relationship** among two or more **Administered Items**

NOTE Metamodel construct is: *Attribute of Administered Item Relationship*.

3.3.7

Administration Record

a collection of administrative information for an **Administered Item**

NOTE Metamodel construct is: *Attribute capsule*.

3.3.8

administrative note

any general note about the **Administered Item**

NOTE Metamodel construct is: *Attribute of Administration Record*.

3.3.9

administrative status

a **designation** of the status in the administrative process of a **Registration Authority** for handling registration requests

NOTE 1 Metamodel construct is: *Attribute of Administration Record*.

NOTE 2 The values and associated meanings of “administrative status” are determined by each **Registration Authority**.
C.f. “**registration status**”.

3.3.10

change description

the description of what has changed in the **Administered Item** since the prior version of the **Administered Item**

NOTE Metamodel construct is: *Attribute of Administration Record.*

3.3.11

Classification Scheme

the descriptive information for an arrangement or division of objects into groups based on characteristics, which the objects have in common

EXAMPLE Origin, composition, structure, application, function, etc.; See ISO/IEC 11179-2.

NOTE Metamodel construct is: *Class.*

3.3.12

classification scheme administration record

the **Administration Record** for a **Classification Scheme**

NOTE Metamodel construct is: *Attribute of Classification Scheme.*

3.3.13

Classification Scheme Item (CSI)

an item of content in a **Classification Scheme**.

NOTE 1 Metamodel construct is: *Class.*

NOTE 2 This may be a node in a taxonomy or ontology, a term in a thesaurus, etc.

3.3.14

Classification Scheme Item Relationship

the **relationship** among items in a **Classification Scheme**

NOTE Metamodel construct is: *Association Class.*

3.3.15

classification scheme item relationship type description

a description of the type of **relationship** between a **Classification Scheme Item** and one or more other **Classification Scheme Items** in a **Classification Scheme**

NOTE Metamodel construct is: *Attribute of Classification Scheme Item Relationship.*

3.3.16

classification scheme item type name

the **name** of the type of the **Classification Scheme Item**

NOTE Metamodel construct is: *Attribute of Classification Scheme Item.*

3.3.17

classification scheme item value

an instance of a **Classification Scheme Item**

NOTE Metamodel construct is: *Attribute of Classification Scheme Item.*

3.3.18

classification scheme membership

the **relationship** of a **Classification Scheme** with its items

NOTE Metamodel construct is: *Relationship*.

3.3.19

classification scheme type name

the **name** of the type of **Classification Scheme**

NOTE Metamodel construct is: *Attribute of Classification Scheme*.

3.3.20

Concept

unit of knowledge created by a unique combination of **characteristics**

NOTE Metamodel construct is: *Class*.

[ISO 1087-1:2000 (3.2.1)]

3.3.21

Concept Relationship

a semantic link among two or more **Concepts**

NOTE Metamodel construct is: *Association Class*.

3.3.22

concept relationship type description

a description of the type of **relationship** among two or more **Concepts**

NOTE Metamodel construct is: *Attribute of Concept Relationship*.

3.3.23

Conceptual Domain (CD)

a set of valid **Value Meanings**

NOTE 1 Metamodel construct is: *Class*.

NOTE 2 The **Value Meanings** may either be enumerated or expressed via a description.

3.3.24

conceptual domain administration record

the **Administration Record** for a **Conceptual Domain**

NOTE Metamodel construct is: *Attribute of Conceptual Domain*.

3.3.25

Conceptual Domain Representation

a **relationship** between a **Conceptual Domain** and a **Value Domain**

NOTE Metamodel construct is: *Relationship*.

3.3.26

Contact

an instance of a role of an individual or an organization (or organization part or organization person) to whom an information item(s), a material object(s) and/or person(s) can be sent to or from in a specified context

NOTE Metamodel construct is: *Class*.

3.3.27

contact information

information to enable a **Contact** to be located or communicated with

NOTE Metamodel construct is: *Attribute of **Contact***.

3.3.28

contact name

the **name** of the **Contact**

NOTE Metamodel construct is: *Attribute of **Contact***.

3.3.29

contact title

the **name** of the position held by the **Contact**

NOTE Metamodel construct is: *Attribute of **Contact***.

3.3.30

Context (for administered item)

a universe of discourse in which a **name** or **definition** is used

NOTE Metamodel construct is: *Class*.

3.3.31

context administration record

the **Administration Record** for a **Context**

NOTE Metamodel construct is: *Attribute of **Context***.

3.3.32

context description

the textual description of the **Context**

NOTE Metamodel construct is: *Attribute of **Context***.

3.3.33

context description language identifier

the **identifier** of the **language** used in the **context description**

NOTE Metamodel construct is: *Attribute of **Context***.

3.3.34

country identifier (of Language Identification)

a country **identifier** further specifying the geographic region associated with the **language**

NOTE 1 Metamodel construct is: *Attribute of **Language Identification***.

NOTE 2 Use the three digit numeric codes from ISO 3166-1, with extensions, if required.

3.3.35

creation date

the date the **Administered Item** was created

NOTE Metamodel construct is: *Attribute of **Administration Record***.

3.3.36

Data Element (DE)

a unit of data for which the **definition**, identification, representation and **Permissible Values** are specified by means of a set of **attributes**

NOTE Metamodel construct is: *Class*.

3.3.37

data element administration record

the **Administration Record** for a **Data Element**

NOTE Metamodel construct is: *Attribute of Data Element*.

3.3.38

Data Element Concept (DEC)

a concept that can be represented in the form of a **Data Element**, described independently of any particular representation

NOTE Metamodel construct is: *Class*.

3.3.39

data element concept administration record

the **Administration Record** for a **Data Element Concept**

NOTE Metamodel construct is: *Attribute of Data Element Concept*.

3.3.40

data element concept conceptual domain relationship

the **relationship** between a **Data Element Concept** and its **Conceptual Domain**

NOTE Metamodel construct is: *Relationship*.

3.3.41

data element concept expression

the **relationship** between a **Data Element** and a **Data Element Concept**

NOTE Metamodel construct is: *Relationship*.

3.3.42

data element concept object class

the **designation** of an **Object Class** for a **Data Element Concept**

NOTE Metamodel construct is: *Attribute of Data Element Concept*.

3.3.43

data element concept property

the **designation** of a **Property** for a **Data Element Concept**

NOTE Metamodel construct is: *Attribute of Data Element Concept*.

3.3.44

Data Element Derivation

the **relationship** among a **Data Element** which is derived, the rule controlling its derivation, and the **Data Element(s)** from which it is derived

NOTE Metamodel construct is: *Association Class*.

3.3.45

Data Element Example

representative illustration of the **Data Element**

NOTE Metamodel construct is: *Class*.

3.3.46

data element example item

actual illustrative case of the **Data Element**

NOTE Metamodel construct is: *Attribute of Data Element Example*.

3.3.47

data element precision

the degree of specificity for a **Data Element**

NOTE 1 Metamodel construct is: *Attribute of Data Element*.

NOTE 2 Expressed as a number of decimal places to be used in any associated **Data Element** values. If not specified, the default precision may be taken from the **unit of measure precision** on the associated **Value Domain**.

3.3.48

data element representation

the **relationship** between a **Data Element** and its **Value Domain**

NOTE Metamodel construct is: *Relationship*.

3.3.49

data element representation class

the class of representation of a **Data Element**

NOTE Metamodel construct is: *Relationship*.

3.3.50

data identifier

the unique **identifier** for an **Administered Item** within a **Registration Authority**

NOTE Metamodel construct is: *Attribute of Item identifier*.

3.3.51

Datatype

a set of distinct values, characterized by properties of those values and by operations on those values

NOTE Metamodel construct is: *Class*.

[ISO/IEC 11404:1996 (4.11)]

3.3.52

datatype annotation

specifying information to further define the **Datatype**

NOTE Metamodel construct is: *Attribute of Datatype*.

3.3.53

datatype description

descriptive information to further clarify the **Datatype**

NOTE Metamodel construct is: *Attribute of Datatype*.

3.3.54

datatype name

a **designation** for the **Datatype**

NOTE Metamodel construct is: *Attribute of Datatype*.

3.3.55

datatype scheme reference

a reference identifying the source of the **Datatype** specification

NOTE 1 In this version of this Standard, the manner of reference is specified by the Registration Authority.

NOTE 2 Metamodel construct is: *Attribute of Datatype*.

3.3.56

Definition (of Administered Item)

the definition of an **Administered Item** within a **Context**

NOTE Metamodel construct is: *Class*.

3.3.57

definition source reference

a reference to the source from which the **Definition** is taken

NOTE Metamodel construct is: *Attribute of Definition*.

3.3.58

definition text

the text of the **Definition**

NOTE Metamodel construct is: *Attribute of Definition*.

3.3.59

derivation input

the **relationship** specifying the source **Data Element**(s) for a **Data Element Derivation**

NOTE Metamodel construct is: *Relationship*.

3.3.60

derivation output

the **relationship** denoting the result of a **Data Element Derivation**

NOTE Metamodel construct is: *Relationship*.

3.3.61

Derivation Rule

the logical, mathematical, and/or other operations specifying derivation

NOTE Metamodel construct is: *Class*.

3.3.62

derivation rule administration record

the **Administration Record** for a **Derivation Rule**

NOTE Metamodel construct is: *Attribute of Derivation Rule*.

3.3.63

derivation rule application

the **relationship** specifying the **Derivation Rule** for a **Data Element Derivation**

NOTE Metamodel construct is: *Relationship*.

3.3.64

derivation rule specification

the text of a specification of **Data Element Derivation**

NOTE Metamodel construct is: *Attribute of Derivation Rule*.

3.3.65

Designation (of Administered Item)

the **designation** of an **Administered Item** within a **Context**

NOTE 1 Metamodel construct is: *Class*.

NOTE 2 See also **3.2.9 designation**.

3.3.66

dimensionality (of Conceptual Domain)

an expression of measurement without units

NOTE 1 Metamodel construct is: *Attribute of Conceptual Domain*.

NOTE 2 ISO 31-0:1982 specifies physical dimensions (e.g. length, mass, velocity). This part of ISO/IEC 11179 also allows non-physical dimensions (e.g. value dimensions such as: currency, quality indicator)

NOTE 3 See also **3.3.131 Unit of Measure**.

3.3.67

documentation language identifier

the **identifier** of the **language** used for documentation by the **Registration Authority**

NOTE Metamodel construct is: *Attribute of Registration Authority*.

3.3.68

effective date

the date an **administered item** became/becomes available to registry users

NOTE Metamodel construct is: *Attribute of Administration Record*.

3.3.69

Enumerated Conceptual Domain

a **Conceptual Domain** that is specified by a list of all its **Value Meanings**

NOTE Metamodel construct is: *Class*.

3.3.70

Enumerated Value Domain

a **Value Domain** that is specified by a list of all its **Permissible Values**

NOTE Metamodel construct is: *Class*.

3.3.71

exemplification

a **relationship** between a **Data Element Example** and its **Data Element**

NOTE Metamodel construct is: *Relationship*.

3.3.72

explanatory comment

descriptive comments about the **Administered Item**

NOTE Metamodel construct is: *Attribute of Administration Record*.

3.3.73

International code designator

the **identifier** of an organization identification scheme

NOTE 1 Metamodel construct is: *Attribute of Registration Authority Identifier*.

NOTE 2 Based on ISO/IEC 6523-1:1998 (3.8) .

NOTE 3 See also ISO/IEC 11179-6.

3.3.74

Item identifier

an **identifier** for an item

NOTE Metamodel construct is: *Attribute capsule*

3.3.75

item registration authority identifier

the **identifier** of the **Registration Authority** registering the item

NOTE 1 Metamodel construct is: *Attribute of Item identifier*.

NOTE 2 See also ISO/IEC 11179-6.

3.3.76

Language Identification

the collection of **identifiers** required to identify a **language** or language variation for a particular purpose

NOTE Metamodel construct is: *Attribute capsule*.

3.3.77

language identifier

information in a **Terminological Entry** which indicates the **name** of a **language**

NOTE 1 Use the three character alphabetic codes and names from ISO 639-2/Terminology, with extensions if required.

NOTE 2 Metamodel construct is: *Attribute of Language Identification*.

3.3.78

Language Section

the part of a **Terminological Entry** containing information related to one **language**

NOTE Metamodel construct is: *Class*.

3.3.79

language section language identifier

the **identifier** of the **language** used to group a set of **Designations** and **Definitions**

NOTE Metamodel construct is: *Attribute of Language Section*.

3.3.80

last change date

the date the **Administered Item** was last changed

NOTE Metamodel construct is: *Attribute of Administration Record.*

3.3.81

name (of Administered Item)

a **name** by which an **Administered Item** is designated within a specific **Context**

NOTE 1 Metamodel construct is: *Attribute of Designation.*

NOTE 2 The general term **name** is defined in sub-clause 3.2.23.

3.3.82

Non-enumerated Conceptual Domain

a **Conceptual Domain** that is not specified by a list of all valid **Value Meanings**

NOTE Metamodel construct is: *Class.*

3.3.83

non-enumerated conceptual domain description

a description of a rule, reference, or range for a set of all **Value Meanings** for the **Conceptual Domain**

NOTE Metamodel construct is: *Attribute of Non-enumerated Conceptual Domain.*

3.3.84

Non-enumerated Value Domain

a **Value Domain** that is specified by a description rather than a list of all **Permissible Values**

NOTE Metamodel construct is: *Class.*

3.3.85

non-enumerated value domain description

a description of a rule, reference, or range for a set of all **Permissible Values** for the **Value Domain**

NOTE Metamodel construct is: *Attribute of Non-enumerated Value Domain.*

3.3.86

Object Class

a set of ideas, abstractions, or things in the real world that are identified with explicit boundaries and meaning and whose properties and behaviour follow the same rules

NOTE Metamodel construct is: *Class.*

3.3.87

object class administration record

the **Administration Record** for an **Object Class**

NOTE Metamodel construct is: *Attribute of Object Class.*

3.3.88

object class qualifier

a qualifier of the **Data Element Concept Object Class**

NOTE Metamodel construct is: *Attribute of Data Element Concept.*

3.3.89

Organization

a unique framework of authority within which a person or persons act, or are designated to act, towards some purpose

NOTE 1 Metamodel construct is: *Class*.

NOTE 2 The kinds of organizations covered by ISO/IEC 6523-1 include the following examples:

- e) an organization incorporated under law;
- f) an unincorporated organization or activity providing goods and/or services including:
 - 1) partnerships;
 - 2) social or other non-profit organizations or similar bodies in which ownership or control is vested in a group of individuals;
 - 3) sole proprietorships
 - 4) governmental bodies .
- g) groupings of the above types of organizations where there is a need to identify these in information interchange.

[ISO/IEC 6523-1:1998 (3.1)]

3.3.90

organization identifier

the **identifier** assigned to an **Organization** within an organization identification scheme, and unique within that scheme

NOTE Metamodel construct is: *Attribute of **Registration Authority Identifier***.

[ISO/IEC 6523-1:1998 (3.10)]

3.3.91

organization mail address

the physical mailing address of the **Organization**

NOTE Metamodel construct is: *Attribute of **Organization***.

3.3.92

organization name

a **designation** for the **Organization**

NOTE Metamodel construct is: *Attribute of **Organization***.

3.3.93

organization part

any department, service or other entity within an organization which needs to be identified for information exchange

[ISO/IEC 6523-1:1998 (3.2)]

3.3.94

organization part identifier (opi)

an **identifier** allocated to a particular **organization part**

NOTE 1 Metamodel construct is: *Attribute of Registration Authority Identifier.*

NOTE 2 See also ISO/IEC 11179-6.

[ISO/IEC 6523-1:1998 (3.11)]

3.3.95
organization part identifier source
the source for the **organization part identifier**

NOTE 1 Metamodel construct is: *Attribute of Registration Authority Identifier.*

NOTE 2 See also ISO/IEC 11179-6.

[Based on ISO/IEC 6523-1:1998 (3.12)]

3.3.96
origin (of Administered Item)
the source (document, project, discipline or model) for the **Administered Item**

NOTE Metamodel construct is: *Attribute of Administration Record.*

3.3.97
Permissible Value
an expression of a **Value Meaning** allowed in a specific **Value Domain**

NOTE Metamodel construct is: *Class.*

3.3.98
permissible value begin date
the date this value became/becomes permissible in the **Value Domain**

NOTE 1 Metamodel construct is: *Attribute of Permissible Value.*

NOTE 2 A Registration Authority may determine whether this date is the date the value becomes valid in a registry or the date the value becomes part of the source domain or some other date.

3.3.99
permissible value end date
the date this value became/becomes no longer permissible in the **Value Domain**

NOTE 1 Metamodel construct is: *Attribute of Permissible Value.*

NOTE 2 A Registration Authority may determine whether this date is the date the value becomes no longer valid in a registry or the date the value becomes no longer part of the source domain or some other date.

3.3.100
permissible value meaning
the **relationship** of a **Value Meaning** from an **Enumerated Conceptual Domain** with a **Permissible Value** from an **Enumerated Value Domain**

NOTE Metamodel construct is: *Relationship.*

3.3.101
permissible value set
the set of **Permissible Values** for an **Enumerated Value Domain**

NOTE Metamodel construct is: *Relationship.*

3.3.102

permitted value

the use of a value as a **Permissible Value** in an **Enumerated Value Domain**

NOTE Metamodel construct is: *Relationship*.

3.3.103

preferred definition

an indicator that the **definition text** is a preferred **definition** for an **Administered Item** within a **language**

NOTE Metamodel construct is: *Attribute of Definition*.

3.3.104

preferred designation

an indicator that the **name** is a preferred term for an **Administered Item** within a **language**

NOTE 1 Metamodel construct is: *Attribute of Designation*.

NOTE 2 See “**main entry term**” in ISO 12620:1999.

3.3.105

Property

a **characteristic** common to all members of an **Object Class**

NOTE Metamodel construct is: *Class*.

3.3.106

property administration record

the **Administration Record** for a **Property**

NOTE Metamodel construct is: *Attribute of Property*.

3.3.107

property qualifier

a qualifier of the **Data Element Concept Property**

NOTE Metamodel construct is: *Attribute of Data Element Concept*.

3.3.108

reference

the **relationship** between a **Reference Document** and an **Administered Item**

NOTE Metamodel construct is: *Relationship*.

3.3.109

Reference Document

a document that provides pertinent details for consultation about a subject

NOTE Metamodel construct is: *Class*.

3.3.110

reference document identifier

an **identifier** for the **Reference Document**

NOTE Metamodel construct is: *Attribute of Reference Document*.

3.3.111

reference document language

the **identifier** of the natural or special **language** used in the **Reference Document**

NOTE Metamodel construct is: *Attribute of Reference Document.*

3.3.112

reference document title

the title of the **Reference Document**

NOTE Metamodel construct is: *Attribute of Reference Document.*

3.3.113

reference document type description

a description of the type of **Reference Document**

NOTE Metamodel construct is: *Attribute of Reference Document.*

3.3.114

reference organization

the **relationship** between a **Reference Document** and an **Organization**

NOTE Metamodel construct is: *Relationship.*

3.3.115

Registrar

a representative of a **Registration Authority**

NOTE Metamodel construct is: *Class.*

3.3.116

registrar identifier

an **identifier** for the **Registrar**

NOTE Metamodel construct is: *Attribute of Registrar.*

3.3.117

registration

the **relationship** between an **Administered Item** and the **Registration Authority**

NOTE Metamodel construct is: *Relationship.*

3.3.118

Registration Authority

an **Organization** responsible for maintaining a register

NOTE Metamodel construct is: *Class.*

3.3.119

registration authority identifier

an **identifier** assigned to a **Registration Authority**

NOTE 1 Metamodel construct is: *Attribute of Registration Authority.*

NOTE 2 See ISO/IEC 11179-6 and ISO/IEC 6523-2.

3.3.120

Registration Authority Identifier
an **identifier** assigned to a **Registration Authority**

NOTE 1 Metamodel construct is: *Class (Attribute capsule)* .

NOTE 2 See ISO/IEC 11179-6 and ISO/IEC 6523-2.

3.3.121

registration authority registrar
the **relationship** between a **Registration Authority** and a **Registrar**

NOTE Metamodel construct is: *Relationship*.

3.3.122

registration status
a **designation** of the status in the registration life-cycle of an **Administered Item**

NOTE 1 Metamodel construct is: *Attribute of Administration Record*.

NOTE 2 Designation values are described in ISO/IEC 11179-6.

3.3.123

Representation Class
the classification of types of representations

NOTE Metamodel construct is: *Class*.

3.3.124

representation class administration record
the **Administration Record** for a **Representation Class**

NOTE Metamodel construct is: *Attribute of Representation Class*.

3.3.125

representation class qualifier
a qualifier to the **Representation Class** used in naming **Data Elements** and **Value Domains**

NOTE Metamodel construct is: *Attribute of Data Element*.

3.3.126

Stewardship (of Administered Item)
the **relationship** of an **Administered Item**, a **Contact**, and an **Organization** involved in the **stewardship** of the **metadata**

NOTE 1 Metamodel construct is: *Association Class*.

NOTE 2 See also **3.2.29 stewardship** (of metadata).

3.3.127

stewardship contact
the contact information for a **Stewardship**

NOTE Metamodel construct is: *Attribute of Stewardship*.

3.3.128

Submission (of Administered Item)

the **relationship** of an **Administered Item**, a **Contact**, and an **Organization** involved in a submission of metadata

NOTE Metamodel construct is: *Association Class*.

3.3.129

submission contact

the contact information associated with a **Submission**

NOTE Metamodel construct is: *Attribute of Submission*.

3.3.130

Terminological Entry

an entry containing information on terminological units for a specific **Administered Item** within a **Context** (subject field)

NOTE Metamodel construct is: *Class*.

3.3.131

Unit of Measure (of Value Domain)

the actual units in which the associated values are measured

NOTE 1 Metamodel construct is: *Class*.

NOTE 2 ISO 31-0:1982 specifies a system of physical measurement (the International System of Units, SI). Physical measurement is only one type of measurement. Value measurement is another type of measurement. This part of ISO/IEC 11179 allows the use of any appropriate system of measurement.

NOTE 3 The **dimensionality** of the associated **Conceptual Domain** must be appropriate for the specified **Unit of Measure**.

3.3.132

unit of measure name

the **name** of a **Unit of Measure**

NOTE Metamodel construct is: *Attribute of Unit of Measure*.

3.3.133

unit of measure precision

the degree of specificity for a **Unit of Measure**

NOTE 1 Metamodel construct is: *Attribute of Unit of Measure*.

NOTE 2 Expressed as a number of decimal places to be used in any associated **Data Element** values. To be used as a default if no precision is specified on the **Data Element** itself.

3.3.134

unresolved issue

any problem that remains unresolved regarding proper documentation of the **Administered Item**

NOTE Metamodel construct is: *Attribute of Administration Record*.

3.3.135

until date

the date an **Administered Item** is no longer effective in the registry

NOTE Metamodel construct is: *Attribute of Administration Record*.

3.3.136

Value

a data value

NOTE Metamodel construct is: *Class*.

3.3.137

Value Domain

a set of **Permissible Values**

NOTE 1 Metamodel construct is: *Class*.

NOTE 2 The **Value Domain** provides representation, but has no implication as to what **Data Element Concept** the **Values** may be associated with nor what the **Values** mean

NOTE 3 The **Permissible Values** may either be enumerated or expressed via a description.

3.3.138

value domain administration record

the **Administration Record** for a **Value Domain**

NOTE Metamodel construct is: *Attribute of Value Domain*.

3.3.139

value domain datatype

the **Datatype** used in a **Value Domain**

NOTE Metamodel construct is: *Attribute of Value Domain*.

3.3.140

value domain format

a template for the structure of the presentation of the **Value(s)**

EXAMPLE – YYYY-MM-DD for a date.

NOTE Metamodel construct is: *Attribute of Value Domain*.

3.3.141

value domain maximum character quantity

the maximum number of characters to represent the **Data Element** value

NOTE 1 Metamodel construct is: *Attribute of Value Domain*.

NOTE 2 Applicable only to character datatypes.

3.3.142

value domain representation class

the class of representation of a **Value Domain**

NOTE Metamodel construct is: *Relationship*.

3.3.143

value domain unit of measure

the unit of measure used in a **Value Domain**

NOTE Metamodel construct is: *Attribute of Value Domain*.

3.3.144

value item

a representation of a **Value Meaning** in a specific **Value Domain** – the actual **Value**

NOTE Metamodel construct is: *Attribute of Value*.

3.3.145

Value Meaning

the meaning or semantic content of a **Value**

NOTE 1 Metamodel construct is: *Class*.

NOTE 2 The representation of **Value Meanings** in a registry shall be independent of (and shall not constrain) their representation in any corresponding **Value Domain**.

3.3.146

value meaning begin date

the effective date of this **Value Meaning** in the **Conceptual Domain**

NOTE 1 Metamodel construct is: *Attribute of Value Meaning*.

NOTE 2 A Registration Authority may determine whether this date is the date the **Value Meaning** becomes valid in a registry or the date the **Value Meaning** becomes part of the source domain or some other date.

3.3.147

value meaning description

a description of a **Value Meaning**

NOTE Metamodel construct is: *Attribute of Value Meaning*.

3.3.148

value meaning end date

the date this **Value Meaning** became/becomes invalid

NOTE 1 Metamodel construct is: *Attribute of Value Meaning*.

NOTE 2 A Registration Authority may determine whether this date is the date the **Value Meaning** becomes no longer valid in a registry or the date the **Value Meaning** becomes no longer part of the source domain or some other date.

3.3.149

value meaning identifier

the unique **identifier** for a **Value Meaning**

NOTE Metamodel construct is: *Attribute of Value Meaning*.

3.3.150

value meaning set

the **relationship** between a **Conceptual Domain** and a set of **Value Meanings**.

NOTE Metamodel construct is: *Relationship*.

3.3.151

version

the unique version **identifier** of the **Administered Item**

NOTE Metamodel construct is: *Attribute of Item identifier*.

3.4 List of Abbreviations

The following abbreviations are defined for use within the subject domain of this document.

3.4.1

CD

Conceptual Domain

3.4.2

CSI

Classification Scheme Item

3.4.3

DE

Data Element

3.4.4

DEC

Data Element Concept

3.4.5

MdR

Metadata Registry

3.4.6

opi

organization part identifier

3.4.7

RA

Registration Authority

3.4.8

VD

Value Domain

4 Structure of a Metadata Registry

4.1 Metamodel for a Metadata Registry

A metamodel is a model that describes other models. A metamodel provides a mechanism for understanding the precise structure and components of the specified models, which are needed for the successful sharing of the models by users and/or software facilities.

This part of ISO/IEC 11179 uses a metamodel to describe the structure of a metadata registry. The registry in turn will be used to describe and model other data, typically about business applications. The registry model is specified as a conceptual data model, i.e. one that describes how relevant information is structured in the natural world. In other words, it is how the human mind is accustomed to thinking of the information.

As a conceptual data model, there need be no one-to-one match between the attributes in the model and fields, columns, objects, et cetera in a database. There may be more than one field per attribute and some entities and relationships may be implemented as fields. There is no intent that an implementation should have a table for each relationship or entity. The metamodel need not be physically implemented as specified.

The structure described by this metamodel may be distributed over several implementations. These implementations may be databases, data repositories, metadata registers, metadata registries, dictionaries, etc.

The model shows constraints on minimum and maximum occurrences of attributes. The constraints on maximum occurrences are to be enforced at all times. The constraints on minimum occurrences are to be enforced when the registration status for the metadata item is "recorded" or higher. In other words, a registration status of "recorded" indicates that all mandatory attributes have been documented.

4.2 Application of the metamodel

Some of the objectives of the metamodel for a metadata registry are to:

- provide a unified view of concepts, terms, value domains and value meanings;
- promote a common understanding of the data described;
- enable the sharing and reuse of the contents of implementations.

A metamodel is necessary for coordination of data representation between persons and/or systems that store, manipulate and exchange data. The metamodel will assist registrars in maintaining consistency among different registries. The metamodel enables systems tools and information registries to store, manipulate and exchange the metadata for data attribution, classification, definition, naming, identification, and registration. In this manner, consistency of data content is assured among systems tools and information registries.

Using the metamodel, mappings to the schema of each tool set can be developed. The metamodel constructs can be translated into the language of each tool set, preserving the concepts represented in the original model.

It is assumed that an implementer will use this conceptual data model to develop a more specific logical data model of the identical sphere of interest. A logical data model describes the same data, but as structured in an information system. It is often referred to as a Model of the Information System. A logical data model can be directly used for database design.

4.3 Specification of the metamodel

When using a model to specify another model, it is easy for the reader to become confused about which model is being referred to at any particular point. To minimize this confusion, this document deliberately uses different terms in the model being specified from those used to do the specification.

The registry metamodel is specified using a subset of the Unified Modelling Language (UML). This document uses the term "metamodel construct" for the model constructs it uses, but "metadata objects" for the model constructs it specifies. The metamodel constructs used are: classes, relationships, attributes, association classes, attribute capsules and encapsulated classes. These terms are defined in clause 3.1, and their use is described in Annex B. The specified metadata objects are defined in clause 3.3, and as the main subject of this clause.

However, there are certain parallels between the two models. For example, the "Object Class" specified in the model is equivalent to the metamodel construct "class", and the "Property" specified in the model is equivalent to the metamodel construct "attribute". The different terms are used to make it clear to which model we refer, not because they represent different concepts. One term that we use at both levels is "datatype", but the level to which it applies should be apparent from the context in which it is used.

4.4 Types and Instances

When considering data and metadata, it is important to distinguish between types of data/metadata, and instances of these types. Clause 4 of this document specifies the types of metadata objects that form the structure of a metadata registry. A metadata registry will be populated with instances of these metadata objects (metadata items), which in turn define types of data, e.g. in an application database. In other words, instances of metadata

specify types of application level data. In turn, the application database will be populated by the real world data as instances of those defined data types.

NOTE ISO/IEC 10027:1990 IRDS Framework explains the concepts of different levels of modelling.

4.5 Extensibility

It is not expected that this metamodel will completely accommodate all users. Particular sectors, such as document management, scientific data, statistical data, require metadata attributes not addressed in this standard. Such extensions shall be considered conformant if they do not violate any of the rules inherent in the structure and content as specified by the metamodel in this standard. Classes, relationships, and attributes may be added to this conceptual data model.

Implementers of this standard may include extensions as part of an implementation, and/or they may provide facilities to allow a registry user to define their own extensions.

4.6 Date References

In this standard, dates are important attributes of an Administration Record and of operations of a registry. There are many different temporal schemata, including calendars. {See ISO 19108:2002} For the purpose of this standard, "date" refers to Gregorian calendar date {See ISO 8601:2000} and the associated default representation is YYYY-MM-DD (i.e. Year-Month-Day). For example, 12 October, 2001 if referenced in numeric form should be 2001-10-12 and not as, for example 12-10-2001 (which might be confused with 10 December, 2001).

Users needing to reference and/or use a temporal schema other than the Gregorian calendar should specify the temporal schema used (e.g. GPS time referencing) utilizing ISO 19108:2002.

For the present, the specification of time in addition to date should be consider a user extension to this standard.

4.7 Description of metamodel

For descriptive purposes, the metamodel is organized into six functional regions:

- Common facilities applying to all administered items (see Figure 1):
 - Administration and Identification (see 4.7)
 - Naming and Definition (see 4.8)
 - Classification (see 4.9)
- Descriptions of specific types of Administered Item:
 - Data Element Concepts (see 4.10)
 - Conceptual and Value Domains (see 4.11)
 - Data Elements (see 4.12).

The division of the model into regions is for descriptive purposes only and has no other significance.

4.7.1 Common facilities

Figure 1 illustrates the relationship of the three common facilities to the administered items in the registry.

The common facilities apply to all administered items as follows:

- Administered Items are identified once only and administered as single items within the registry.
- Administered Items are named and defined in at least one Context, and possibly in multiple Contexts. Within each Context, Names and Definitions may be specified in one or more languages.
- Administered Items may be classified in zero or more Classification Schemes.

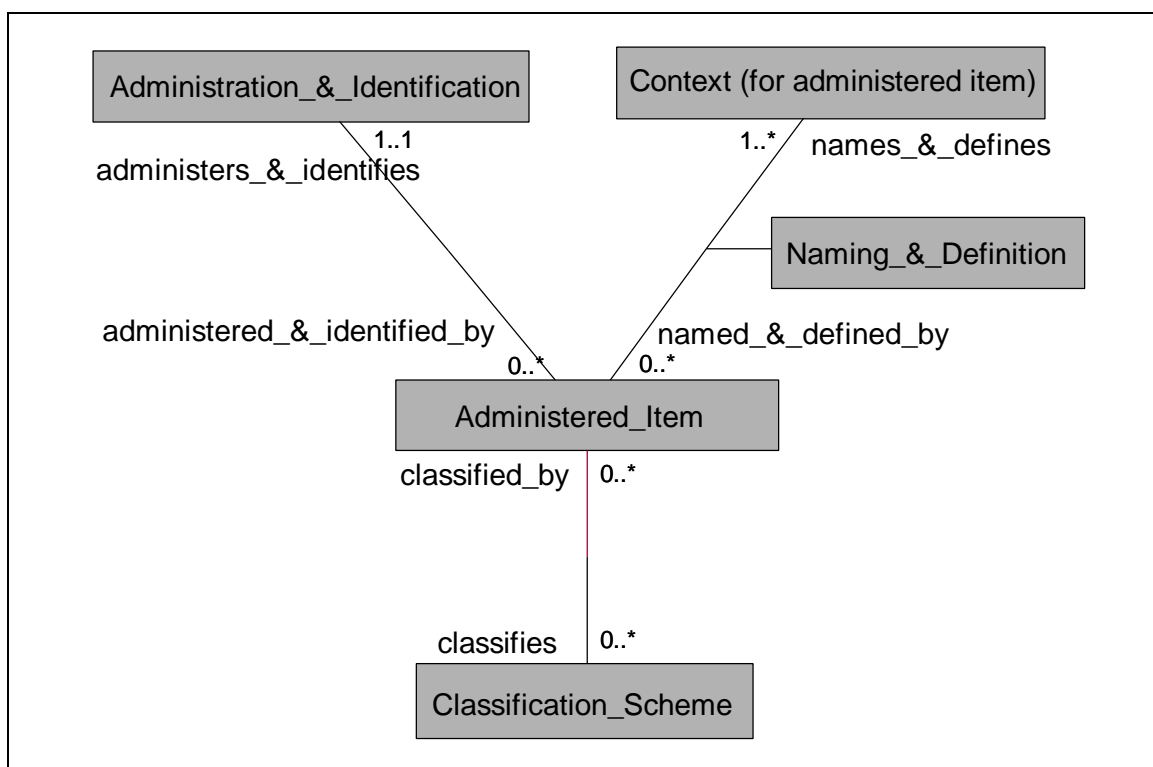


Figure 1: Common facilities for all Administered Items

4.7.2 Types of Administered Items

This part of ISO/IEC 11179 specifies the following types of Administered Items, as listed in Figure 2. The Administered Items shown in the figure are described in more detail later in this clause. Additional types of Administered Item may be defined as extensions to this standard.

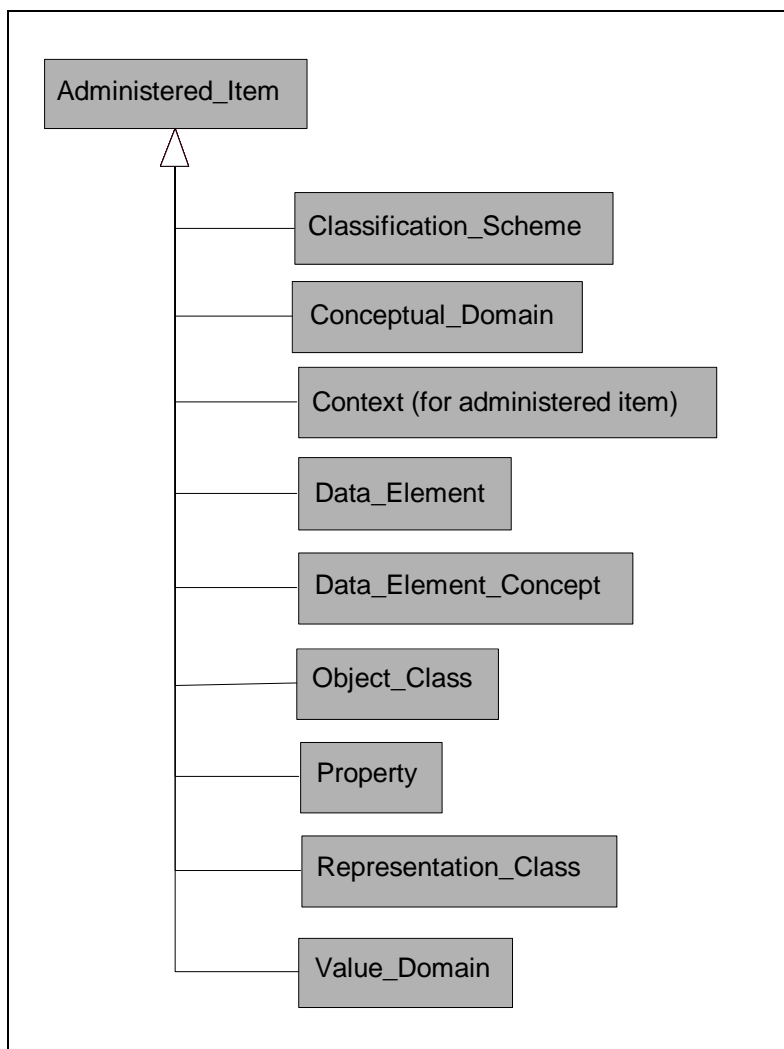


Figure 2: Types of Administered Items

4.8 Administration and Identification region

The Administration and Identification region supports the administrative aspects of administered items in a registry. This region addresses:

- the identification and registration of items submitted to the registry
- the organizations that have submitted items to the registry, and/or that are responsible for items within the registry, including Registration Authorities
- contact information for organizations
- supporting documentation
- relationships among administered items.

4.8.1 Metadata objects in the Administration and Identification region

Figure 3 shows the classes, relationships, attribute capsules and attributes that support Administration and Identification. Figure 4 shows the encapsulated classes used as datatypes on attribute capsules.

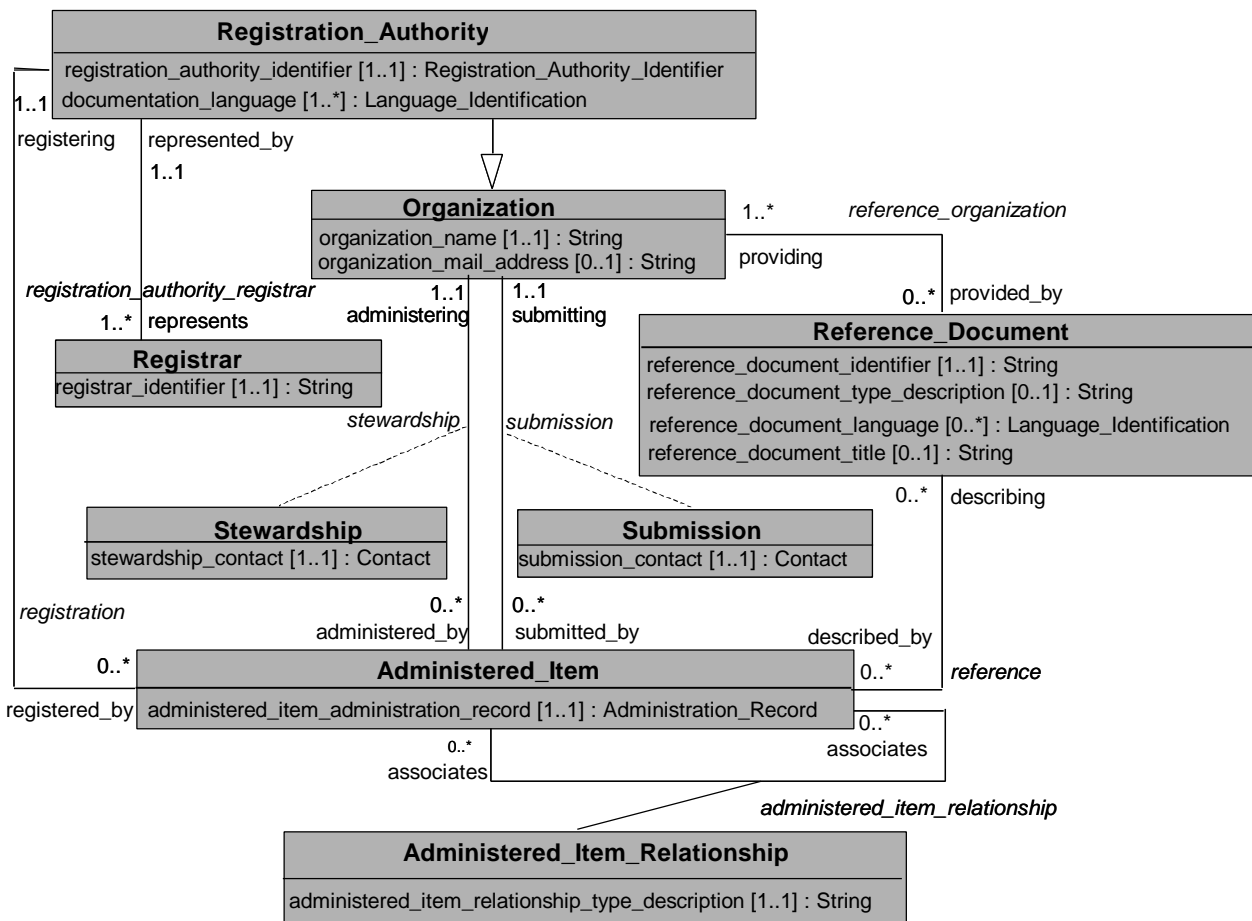


Figure 3: Administration and identification metamodel region

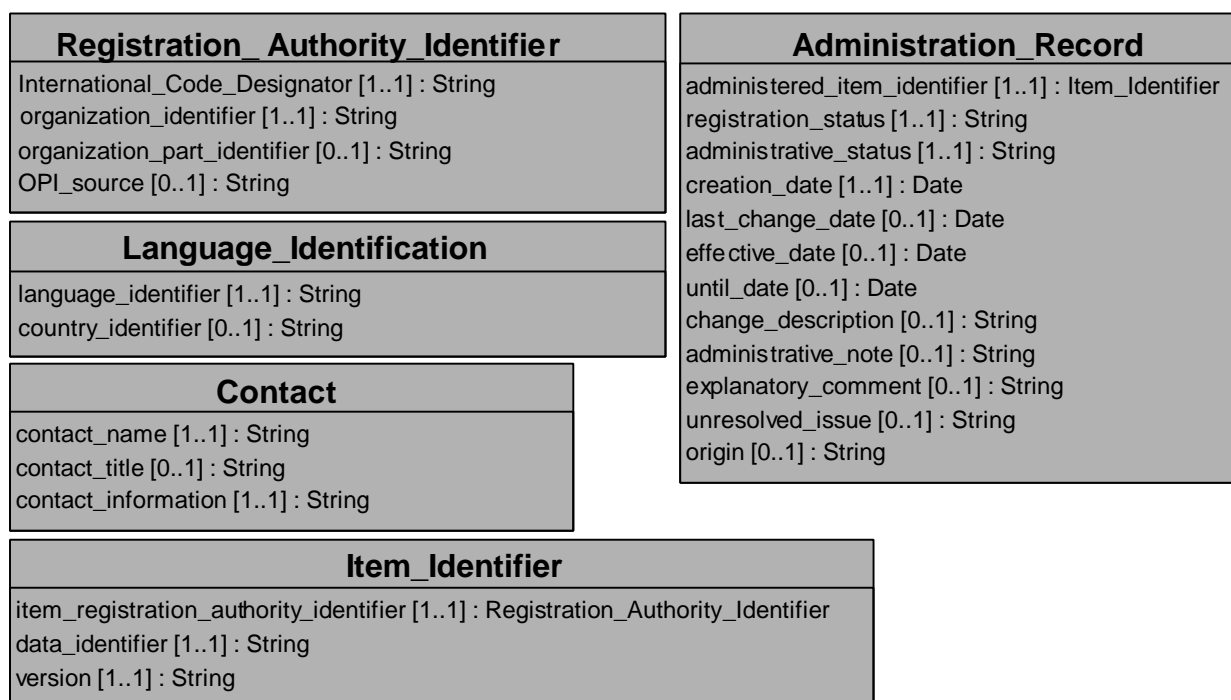


Figure 4: Administration and identification region – Encapsulated classes used as datatypes

4.8.1.1 Administered Item

An *Administered Item* may be any one of those types listed in Figure 2. Each instance of an *Administered Item* encapsulates its own *Administration Record*. An *Administered Item* is submitted by an *Organization* represented by the relationship *submission* in Figure 3. An *Administered Item* is registered by a *Registration Authority* represented by the relationship *registration* in Figure 3. An *Administered Item* is administered by an *Organization* represented by the relationship *stewardship* in Figure 3. An *Administered Item* may be described by zero or more *Reference Documents* as represented by the relationship *reference* in Figure 3.

Each instance of an *Administered Item* through its *Administration Record* shall have a unique *administered item identifier* used to identify it and that distinguishes it from any other *Administered Item*. Each *Administered Item* in the owner's metadata registry shall have (as part of the *Administration Record*):

- a *registration status* indicating the point in a registration life cycle applying to it
- an *administrative status* indicating the point in the *Registration Authority's* registration process.

<u>Attribute name</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>Administered item</i> <i>administration record</i>	One per <i>Administered Item</i> .	Administration_Record

4.8.1.2 Administered Item Relationship

Two or more *Administered Items* may be associated through an *Administered Item Relationship*. Each *Administered Item Relationship* has exactly one *administered item relationship type description*. The associated *Administered Items* may be registered by the same or different *Registration Authorities*.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>administered item relationship type description</i>	One per <i>Administered Item Relationship</i>	String

4.8.1.3 Administration Record

An instance of an *Administration Record* records information about an *Administered Item* in the *Registry*. The *Administration Record* instance provides a basis for identifying, naming, defining, classifying and recording administrative information about the *Administered Item* in the *Registry*.

When an *Administered Item* is modified, it becomes a new *version* of the *Administered Item* and it thus requires a new *version* of its *Administration Record*. The *administration record - creation date*, the reason for change (*administration record - change*), the *contact persons* for the responsible and submitting organizations, *Registration Authority*, and the *Registrar* shall be provided for this new *Administered Item*. The *Registrar* may collect history by retaining the old *Administration Record*.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>administered item identifier</i>	One per <i>Administration record</i>	Item_Identifier
<i>administrative note</i>	Zero or one per <i>Administration record</i>	String
<i>administrative status</i>	One per <i>Administration record</i>	String
<i>change description</i>	One per <i>Administration record</i> conditional on presence of last change date	String
<i>creation date</i>	One per <i>Administration record</i>	Date
<i>effective date</i>	Zero or one per <i>Administration record</i>	Date
<i>explanatory comment</i>	Zero or one per <i>Administration record</i>	String
<i>last change date</i>	Zero or one per <i>Administration record</i>	Date
<i>origin</i>	Zero or one per <i>Administration record</i>	String
<i>registration status</i>	One per <i>Administration record</i>	String
<i>unresolved issue</i>	Zero or one per <i>Administration record</i>	String
<i>until date</i>	Zero or one per <i>Administration record</i>	Date

4.8.1.4 Contact

The encapsulated class *Contact* is used to specify the contact information for *stewardship contact* and *submission contact*.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>contact information</i>	One per <i>Contact</i> .	String
<i>contact name</i>	One per <i>Contact</i> .	String
<i>contact title</i>	Zero or one per <i>Contact</i> .	String

4.8.1.5 Item Identifier

The encapsulated class *Item Identifier* is used to specify the unique identifier for an *Administered Item*. The *Item Identifier* is composed of three parts as shown in the following attribute list. The *item registration authority identifier* identifies the owning *Registration Authority*. The *data identifier* uniquely identifies an *Administered Item* within a *Registration Authority*. *Data identifiers* shall be unique within a *Registration Authority* for each occurrence of an *Administered Item*. The *version* is used to distinguish multiple instances of the same *Administered Item* as it undergoes changes.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>item registration authority identifier</i>	One per <i>Item identifier</i>	Registration Authority Identifier
<i>data identifier</i>	One per <i>Item identifier</i>	String
<i>version</i>	One per <i>Item identifier</i>	String

4.8.1.6 Language Identification

The encapsulated class *Language Identification* serves as an identifier for a language. It is used in:

- the *Registration Authority* class to identify the default language(s) of the registration authority
- the *Reference Document* class to identify the language(s) used within the document
- the *Language Section* class of the Naming and Definition region to identify the language used for terms and definitions within that section.

The identifier comprises a mandatory *language identifier* and an optional *country identifier*, the latter being used to distinguish variations in language use in different countries.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>language identifier</i>	One per <i>Language identification</i>	String NOTE Use the three character alphabetic codes from ISO 639-2/Terminology, with extensions if required.
<i>country identifier</i>	Zero or one per <i>Language identification</i>	String NOTE Use the three digit numeric codes from ISO 3166-1, with extensions if required.

4.8.1.7 Organization

An *Organization* can play one or more roles with respect to a Metadata Registry. The roles currently recognized in this part of ISO/IEC 11179 are: *Registration Authority*, *reference organization*, *steward* (of an *Administered Item*) – represented by the relationship *stewardship* – and *submitter* (of an *Administered Item*) – represented by the relationship *submission*.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>organization name</i>	One per <i>Organization</i>	String
<i>organization mail address</i>	Zero or one per <i>Organization</i>	String

4.8.1.8 Reference Document

An *Administered Item* may be *described by one or more Reference Documents*. For each *Reference Document*, the *Organization* that originated the *Reference Document* must be identified, as shown by the relationship *reference organization* in Figure 3.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>reference document identifier</i>	One per <i>Reference Document</i>	String
<i>reference document language identifier</i>	From zero to many per <i>Reference document</i> (absence of a language indicates use of the same language as specified by <i>Registration Authority documentation language identifier</i>)	Language_Identification
<i>reference document title</i>	Zero or one per <i>reference document</i>	String
<i>reference document type description</i>	Zero or one for each <i>reference document</i>	String

4.8.1.9 Registrar

A *Registration Authority* is represented by one or more *Registrars* — these are the persons who perform the administrative steps to register *Administered Items* in a metadata registry.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>registrar identifier</i>	One for each <i>registrar</i> in a <i>registration authority</i>	String

4.8.1.10 Registration Authority

A *Registration Authority* is any *Organization* authorized to register *metadata*. A *Registration Authority* is a subtype of *Organization* and inherits all of its attributes and relationships. An *Administered Item* has a *Registration Authority* that is its owner, shown by the relationship *registration* in Figure 3. A *Registration Authority* may register many *Administered Items*.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>registration authority identifier</i>	One per <i>registration authority</i>	<i>Registration Authority Identifier</i>
<i>documentation language identifier</i>	From one to many per <i>registration authority</i>	<i>Language Identification</i>

4.8.1.11 Registration Authority Identifier

The encapsulated class *Registration Authority Identifier* is used to uniquely identify a *Registration Authority*. The sources of values for each part of the identifier are specified in ISO/IEC 11179-6.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>international code designator</i>	One per <i>registration authority identifier</i>	String
<i>organization identifier</i>	One per <i>registration authority identifier</i>	String
<i>organization part identifier (OPI)</i>	One per <i>registration authority identifier</i>	String
<i>OPI source</i>	One per <i>registration authority identifier</i>	String

4.8.1.12 Stewardship

An *Organization* shall be identified as the steward responsible for *administering* each *Administered Item*, as represented by the relationship *Stewardship* in Figure 3. This relationship identifies a *stewardship contact* for the *Administered Item*.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>stewardship contact</i>	One per <i>Stewardship</i>	Contact

4.8.1.13 Submission

For each *Administered Item*, an *Organization* shall be identified as the submitter as represented by the relationship *Submission* in Figure 3. This relationship identifies a *submission contact* for the *Administered Item*

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>submission contact</i>	One per <i>Submission</i>	Contact

4.9 Naming and Definition Region

The Naming and Definition region is used to manage the names and definitions of administered items and the contexts that provide the sphere for the names. It is recognized that an administered item may have many names that will vary depending on discipline, locality, technology, etc.

4.9.1 Metadata objects in the Naming and Definition region

Figure 5 represents the Naming and Definition region. This region of the metamodel is based on, and is consistent with, terminological models developed by ISO TC37.

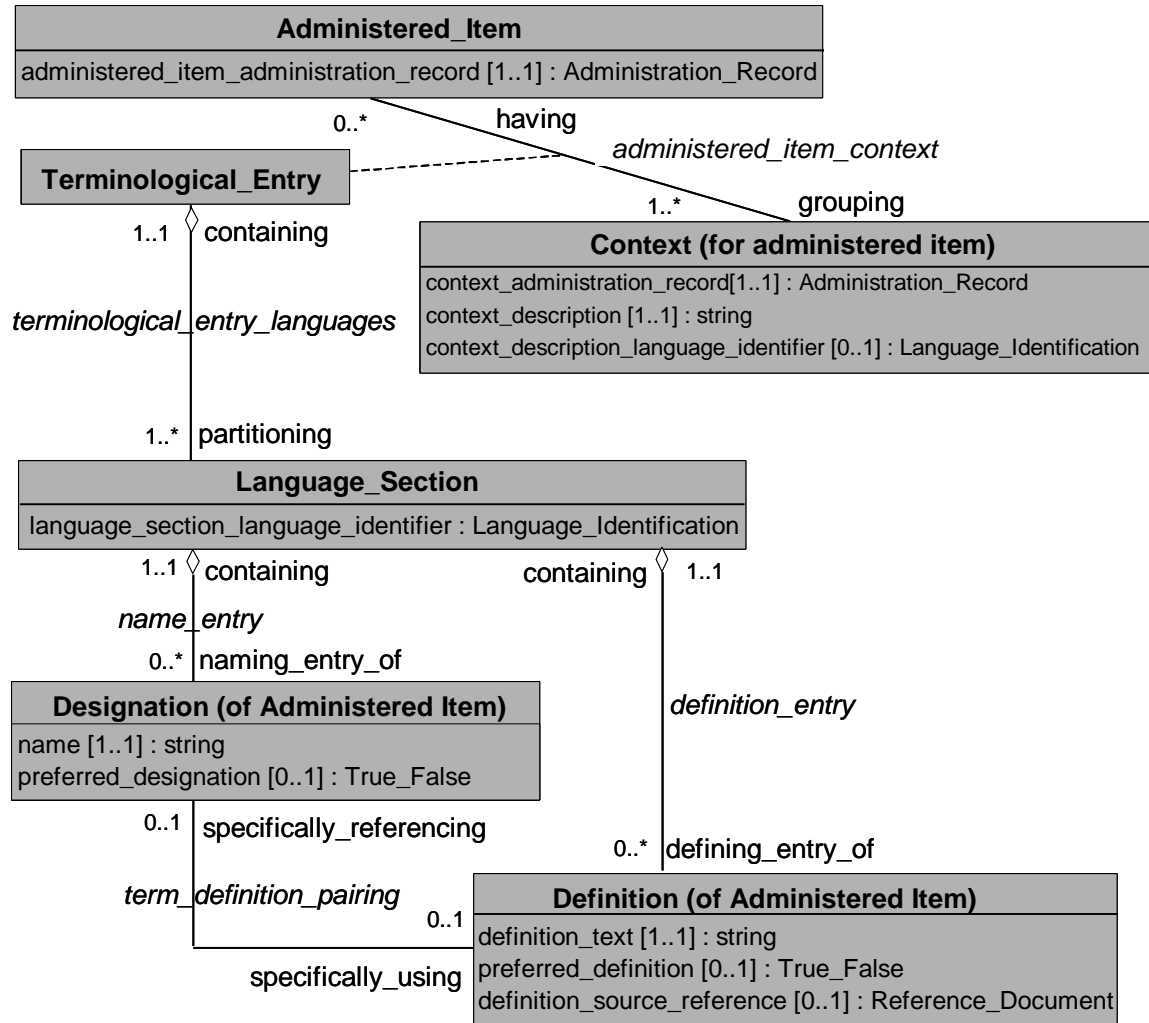


Figure 5: Naming and Definition metamodel region

ISO/IEC 11179-5 *Naming and definition principles for data elements* provides rules and guidelines for naming and definition of administered items within a context.

The encapsulated classes used as datatypes on attribute capsules are described in sub-clause 4.7.1 and Figure 4, since they are also used in the Administration and Identification region.

4.9.1.1 Administered Item

Administered Item is described in sub-clause 4.7.1.1.

4.9.1.2 Context

Each *Administered Item* is named and defined within one or more *Contexts*. A *Context* defines the scope within which the subject data has meaning. A *Context* may be a business domain, an information subject area, an information system, a database, file, data model, standard document, or any other environment determined by the owner of the registry. Each *Context* is itself managed as an *Administered Item* within the registry and is given a *name* and a *definition*.

NOTE The *Context* within which a *Context* is named and defined will probably be the registry itself, but could be broader, and could simply be specified as being this International Standard.

For each *Context* with which an *Administered Item* is associated through a *Terminological Entry*, the *Administered Item* must have at least one *Designation (name)* and at least one *Definition*.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>context administration record</i>	One per <i>Context</i>	<i>Administration Record</i>
<i>context description</i>	One per <i>Context</i>	String
<i>context description language identifier</i>	Zero or one per <i>Context</i>	<i>Language Identification</i>

4.9.1.3 Terminological Entry

A *Terminological Entry* applies to an *Administered Item* in a particular *Context*. It provides a grouping of *Designations* and *Definitions* partitioned into *Language Sections*, allowing the *Administered Item* to be named and defined within the *Context* in multiple languages.

An *Administered Item* may have one or more *Terminological Entries*, each in a particular *Context*. Each *Terminological Entry* contains one or more *Language Sections* as represented by the relationship *terminological entry languages* in Figure 5.

4.9.1.4 Language Section

If a registry supports multiple languages, the language(s) associated with particular names and definitions need to be identified. A *Language Section* partitions a *Terminological Entry* by *Language*, as represented by the relationship *terminological entry languages* in Figure 6. A *language section language identifier* identifies the *Language* associated with a particular *Language Section*. A *Language Section* contains zero or more *Designations* as represented by the relationship *name entry* in Figure 6. A *Language Section* contains zero or more *Definitions* as represented by the relationship *definition entry* in Figure 5.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>language section language identifier</i>	One per <i>Language section</i> .	<i>Language Identification</i>

4.9.1.5 Definition

The *Definition* class provides the *defining entry* of a *Language Section* in the *Terminological Entry* for an *Administered Item* in a particular *Context*. In other words, it is where the definition for an *Administered Item* is specified in a particular language for a particular *Context*. Where multiple *Definitions* are provided within the same *Language Section*, one of them may be specified as the *preferred definition*.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>definition text</i>	One per <i>definition</i>	String
<i>definition source reference</i>	Zero or one per <i>Definition</i>	<i>Reference Document</i>
<i>preferred definition</i>	Zero or one per <i>Definition</i>	True_False

4.9.1.6 Designation

The *Designation* class provides the *naming entry* of a *Language Section* in the *Terminological Entry* for an *Administered Item* in a particular *Context*. In other words, it is where the name for an *Administered Item* is specified in a particular language for a particular *Context*. Where multiple *Designations* are provided within the same *Language Section*, one of them may be specified as the *preferred designation*.

<u>Attribute</u>	<u>Occurrences</u>	<u>Datatype</u>
<i>name</i>	One per <i>Designation</i>	String
<i>preferred designation</i>	Zero or one per <i>Designation</i>	True_False

4.10 Classification Region

Figure 6 represents the Classification region.

The Classification region provides a facility to register and administer *Classification Schemes* and their constituent *Classification Scheme Items*. Optionally, a *Classification Scheme* may be used to classify *Administered Items* within the registry. Some *Classification Schemes* will be more applicable to classifying objects in the real world than they will be to classifying metadata objects in a registry.

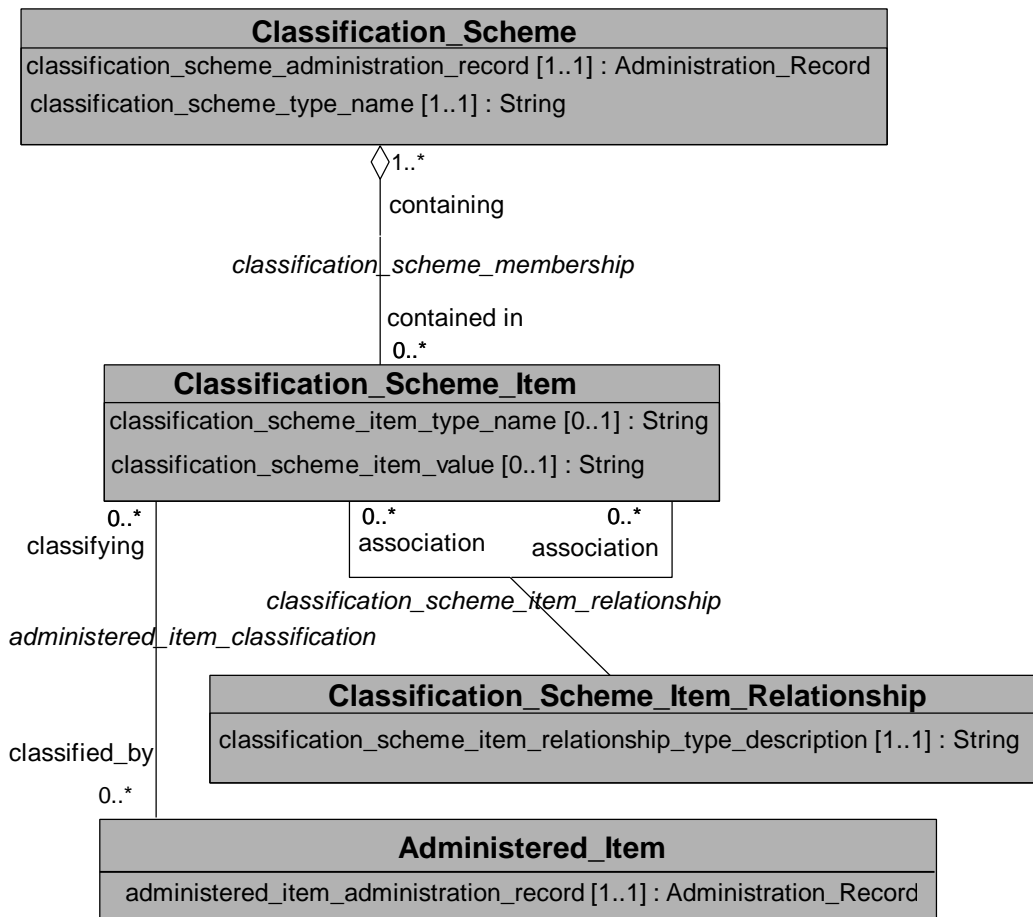


Figure 6: Classification metamodel region

4.10.1 Metadata Objects in the Classification region

4.10.1.1 Administered Item

Administered Item is described in sub-clause 4.7.2.1.

An *Administered Item* may be classified in one or more *Classification Schemes*, by associating it with one or more *Classification Scheme Items* as represented by the relationship *administered item classification* in Figure 6. Such classification is optional.

4.10.1.2 Classification Scheme

A *Classification Scheme* may be a taxonomy, a network, an ontology, or any other terminological system. 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.

A *Classification Scheme* is a sub-type of *Administered Item*, inheriting its attributes and relationships, which allows it to be identified, named, defined and optionally classified.

An *Administered Item* is named within a specific *Context*, and may have different *names* in different *Contexts*. As an *Administered Item* itself, a *Classification Scheme* is also named within one or more *Contexts*. For an *Administered Item* to be considered to have a *name* within a *Classification Scheme*, the *Administered Item* and the *Classification Scheme* must share a common *Context*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>classification scheme</i> <i>administration record</i>	One per <i>Classification Scheme</i>	<i>Administration Record</i>
<i>classification scheme type</i> <i>name</i>	One per <i>Classification Scheme</i>	String

4.10.1.3 Classification Scheme Item

A *Classification Scheme Item* represents an individual item within a *Classification Scheme*. The *Classification Scheme Item* may have either a *classification scheme item type name*, a *classification scheme item value*, or both.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>classification scheme item</i> <i>type name</i>	Zero or one per <i>Classification Scheme Item</i>	String
<i>classification scheme item</i> <i>value</i>	Zero or one per <i>Classification Scheme Item</i>	String

4.10.1.4 Classification Scheme Item Relationship

A *Classification Scheme Item Relationship* associates two or more *Classification Scheme Items* within a *Classification Scheme*. Such relationships serve to assist navigation through a large number of *Classification Scheme Items*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>classification scheme item</i> <i>relationship type description</i>	One per <i>Classification Scheme Item Relationship</i>	String

4.11 Data Element Concept Region

The data element concept region is illustrated in Figure 7. The purpose of the data element concept region is to maintain the information on the concepts upon which the data elements are developed. The metadata objects in this region concentrate on semantics. The concepts are independent of any internal or external physical representation. The metadata objects in this region are Object Classes (encompassing Concepts and Concept Relationships) and Properties, which may be combined to form Data Element Concepts.

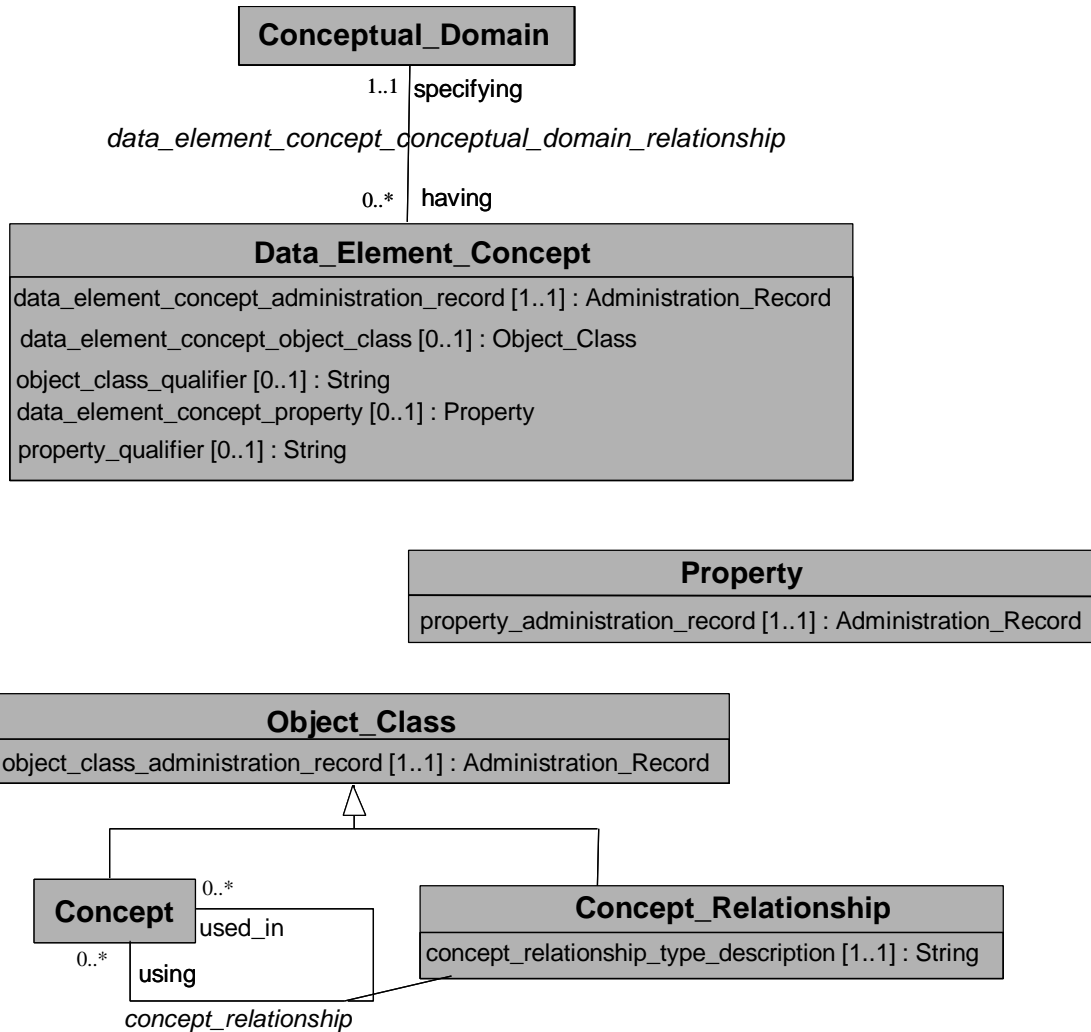


Figure 7: Data Element Concept metamodel region

4.11.1 Metadata objects in the Data Element Concept region

4.11.1.1 Object Class

An *Object Class* is a set of ideas, abstractions, or things in the real world that can be identified with explicit boundaries and meaning and whose properties and behavior follow the same rules. It may be either a single or a group of associated concepts, abstractions, or things. An *Object Class* may be a single unit of thought (i.e., *Concept*) or a set of *Concepts* in a relationship with each other to form a more complex concept (i.e., *Concept Relationship*). A *Concept* and a *Concept Relationship* are subtypes of an *Object Class*. Each *Concept Relationship* carries a *concept relationship type description* that describes the nature of the relationship.

As an *Administered Item*, an *Object Class* carries its own *Administration Record* information, allowing it to be identified, named, defined and optionally classified within a *Classification Scheme*. An *Object Class* may be associated with other *Administered Items*, including other *Object Classes*, via the *Administered Item Relationship* shown in Figure 3 in the Administration and Identification Region. The nature of the relationship is described using the *administered item relationship type description*. An *Object Class* may be registered as an *Administered Item* without necessarily being associated with a *Data Element Concept* or, through the latter, a *Property*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>object class administration record</i>	One per <i>Object Class</i>	<i>Administration_Record</i>
<i>concept relationship type description</i>	One per <i>Concept Relationship</i>	String

4.11.1.2 Property

A *Property* is a characteristic common to all members of an *Object Class*. It may be any feature that humans naturally use to distinguish one individual object from another. It is the human perception of a single characteristic of an *Object Class* in the real world. It is conceptual and thus has no particular associated means of representation by which the *Property* can be communicated.

As an *Administered Item*, a *Property* carries its own *Administration Record* information, allowing it to be identified, named, defined and optionally classified within a *Classification Scheme*. A *Property* may be associated with other *Administered Items*, including other *Properties*, via the *Administered Item Relationship* shown in Figure 3 in the Administration and Identification Region. The nature of the relationship is described using the *administered item relationship type description*. A *Property* may be registered as an *Administered Item* without necessarily being associated with a *Data Element Concept* or, through the latter, an *Object Class*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>property administration record</i>	One per <i>Property</i>	<i>Administration_Record</i>

4.11.1.3 Data Element Concept

A *Data Element Concept* is a concept that can be represented in the form of a data element, described independently of any particular representation. A *Data Element Concept* may have zero or one *Object Class* and zero or one *Property*. The union of a *Property* and an *Object Class* provides significance beyond either that of the *Property* or the *Object Class*. A *Data Element Concept* thus has a *Definition* independent from the *Definition* of the *Object Class* or the *Property*.

As an *Administered Item*, a *Data Element Concept* carries its own *Administration Record* information, allowing it to be identified, named, defined and optionally classified within a *Classification Scheme*. A *Data Element Concept* may be associated with other *Administered Items*, including other *Data Element Concepts*, via the *Administered Item Relationship* shown in Figure 3 in the Administration and Identification Region. The nature of the relationship is described using the *administered item relationship type description*.

A *Data Element Concept* may be registered as an *Administered Item* without necessarily being associated with any *Data Element*, but a *Data Element Concept* shall be associated with exactly one *Conceptual Domain*. The *Conceptual Domain* specifies all valid *Value Meanings* of a *Data Element Concept*. The *Conceptual Domain* is described in the Conceptual and Value Domain Region sub-clause that follows this sub-clause.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>data element concept administration record</i>	One per <i>Data Element Concept</i>	<i>Administration_Record</i>
<i>data element concept object class</i>	Zero or one per <i>Data Element Concept</i>	<i>Object_Class</i>
<i>data element concept property</i>	Zero or one per <i>Data Element Concept</i>	<i>Property</i>
<i>object class qualifier</i>	Zero or one per <i>Data Element Concept</i>	String
<i>property qualifier</i>	Zero or one per <i>Data Element Concept</i>	String

4.12 Conceptual and Value Domain Region

This region of the metamodel addresses the administration of *Conceptual Domains* and *Value Domains*. These domains can be viewed as logical code sets and physical code sets. *Conceptual Domains* support *Data Element Concepts* and *Value Domains* support *Data Elements*. The region is illustrated in Figure 8.

4.12.1 Metadata objects in the Conceptual and Value Domain region

4.12.1.1 Conceptual Domain

A *Conceptual Domain* is a set of *Value Meanings*, which may either be enumerated or expressed via a description.

As an *Administered Item*, a *Conceptual Domain* carries its own *Administration Record* information, allowing it to be identified, named, defined and optionally classified within a *Classification Scheme*.

A *Conceptual Domain* may be associated with other *Administered Items*, including other *Conceptual Domains*, via the *Administered Item Relationship* shown in Figure 3 in the Administration and Identification Region. The nature of the relationship is described using the *administered item relationship type description*. Through the *Administered Item Relationship*, a *Conceptual Domain* may be composed of other *Conceptual Domains* or may be a member (component) of a larger *Conceptual Domain*.

A *Conceptual Domain* may specify a constraint such as "linear measure" as its dimensionality. When a dimensionality is specified, any *Value Domain* that is based on this *Conceptual Domain* shall specify a Unit of Measure that is consistent with this dimensionality.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>conceptual domain administration record</i>	One per <i>Conceptual Domain</i>	<i>Administration_Record</i>
<i>dimensionality</i>	Zero or one per <i>Conceptual Domain</i>	String

4.12.1.2 Enumerated Conceptual Domain

A *Conceptual Domain* sometimes contains a finite allowed inventory of notions that can be enumerated. Such a *Conceptual Domain* is referred to as an *Enumerated Conceptual Domain*. An example of an *Enumerated Conceptual Domain* is the notion of countries that is specified in ISO 3166, Codes for the representation of names of countries. As a sub-type of *Conceptual Domain*, an *Enumerated Conceptual Domain* inherits the attributes and relationships of the former.

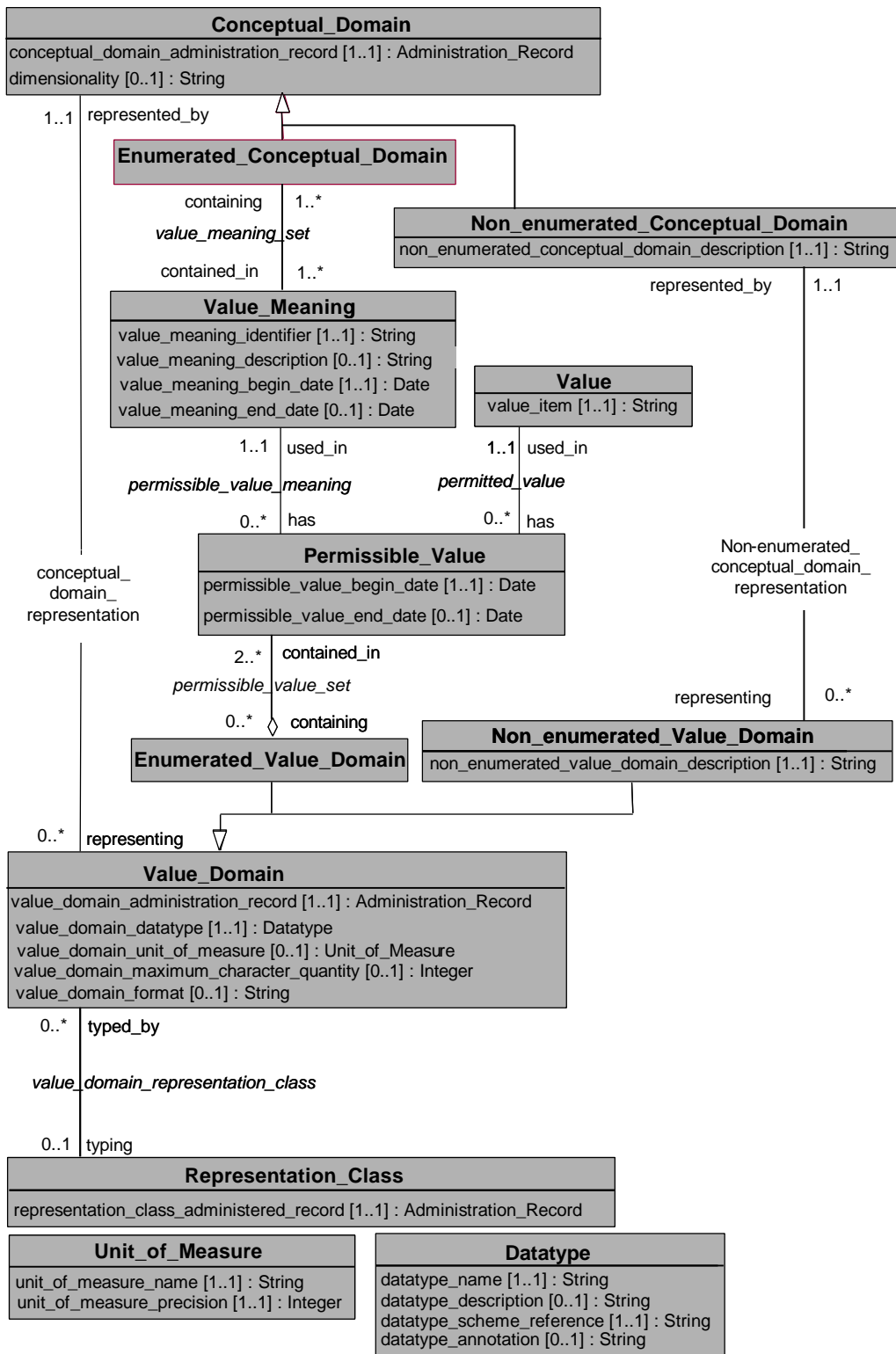


Figure 8: Conceptual and value domain metamodel region

4.12.1.3 Value Meanings

Each member of an *Enumerated Conceptual Domain* has a *Value Meaning* that provides its distinction from other members. In the example of ISO 3166, the notion of each country as specified would be the *Value Meanings*. The representation of *Value Meanings* in a registry shall be independent of (and shall not constrain) their representation in any corresponding *Value Domain*. A particular *Value Meaning* may have more than one means of representation by *Permissible Values*—each from a distinct *Enumerated Value Domain*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>value meaning identifier</i>	One per <i>Value Meaning</i>	String
<i>value meaning begin date</i>	One per <i>Value Meaning</i>	Date
<i>value meaning description</i>	Zero or one per <i>Value Meaning</i>	String
<i>value meaning end date</i>	Zero or one per <i>Value Meaning</i>	Date

4.12.1.4 Non-enumerated Conceptual Domain

A *Conceptual Domain* that cannot be expressed as a finite set of *Value Meanings* is called a *Non-enumerated Conceptual Domain*. It may be expressed via a description, such as a rule, a procedure, or a range (i.e., interval). As a sub-type of *Conceptual Domain*, a *Non-enumerated Conceptual Domain* inherits the attributes and relationships of the former.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>non-enumerated conceptual domain description</i>	One per <i>Non-enumerated Conceptual Domain</i>	String

4.12.1.5 Value Domain

One of the key components of a representation is the *Value Domain*. A *Value Domain* provides representation, but has no implication as to what *Data Element Concept* the values are associated nor what the values mean.

A *Value Domain* is associated with a *Conceptual Domain*. A *Value Domain* provides a representation for the *Conceptual Domain*. An example of a *Conceptual Domain* and a set of *Value Domains* is ISO 3166, Codes for the representation of names of countries. For instance, ISO 3166 describes the set of seven *Value Domains*: short name in English, official name in English, short name in French, official name in French, alpha-2 code, alpha-3 code, and numeric code.

As an *Administered Item*, a *Value Domain* carries its own *Administration Record* information, allowing it to be identified, named, defined and optionally classified in a *Classification Scheme*.

A *Value Domain* may be associated with other *Administered Items*, including other *Value Domains*, via the *Administered Item Relationship* shown in Figure 3 in the Administration and Identification Region. The nature of the relationship is described using the *administered item relationship type description*. Through the *Administered Item Relationship*, a *Value Domain* may be composed of other *Value Domains* or may be a member (component) of a larger *Value Domain*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>value domain administration record</i>	One per <i>Value Domain</i>	<i>Administration_Record</i>
<i>value domain datatype</i>	One per <i>Value domain</i>	<i>Datatype</i>
<i>value domain format</i>	Zero or one per <i>Value Domain</i>	String
<i>value domain maximum character quantity</i>	Zero or one per <i>Value Domain</i>	Integer
<i>value domain unit of measure</i>	Zero or one per <i>Value Domain</i>	<i>Unit_of_Measure</i>

4.12.1.6 Enumerated Value Domain

An *Enumerated Value Domain* is one where the *Value Domain* is expressed as an explicit set of two or more *Permissible Values*. As a sub-type of *Value Domain*, an *Enumerated Value Domain* inherits the attributes and relationships of the former.

4.12.1.7 Permissible Value

A *Permissible Value* is an expression of a *Value Meaning* within an *Enumerated Value Domain*. It is one of a set of such values that comprises an *Enumerated Value Domain*. Each *Permissible Value* is associated with a *Value Meaning*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>permissible value begin date</i>	One per <i>Permissible Value</i>	Date
<i>permissible value end date</i>	Zero or one per <i>Permissible Value</i>	Date

4.12.1.8 Value

This is the actual value associated with a *Permissible Value* in an *Enumerated Value Domain*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>value item</i>	One per <i>Value</i>	String

4.12.1.9 Non-enumerated Value Domain

A *Value Domain* may be expressed via a description, such as a rule, a procedure, or a range (i.e., interval), rather than as an explicit set of *Permissible Values*. Such a *Value Domain* is called a *Non-enumerated Value Domain*. As a sub-type of *Value Domain*, a *Non-enumerated Value Domain* inherits the attributes and relationships of the former.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>non-enumerated value domain description</i>	One per <i>Non-enumerated Value Domain</i>	String

4.12.1.10 Datatype

A *Value Domain* is associated with a *Datatype* — a set of distinct values, characterized by properties of those values and by operations on those values, for example the category used for the collection of letters, digits, and/or symbols to depict values of a *Data Element* determined by the operations that may be performed on the *Data Element*.

A *Datatype* is designated by a *data type name*, and described by a *datatype description*. The *datatype name* is usually drawn from some external source, which is designated by a *datatype scheme reference*. Additional information may optionally be provided using the *datatype annotation*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>datatype name</i>	One per <i>Datatype</i>	String
<i>datatype description</i>	One per <i>Datatype</i>	String
<i>datatype scheme reference</i>	One per <i>Datatype</i>	String
<i>datatype annotation</i>	Zero or one per <i>Datatype</i>	String

4.12.1.11 Unit of Measure

If meaningful, a *Value Domain* may be associated with a *Unit of Measure* — the unit in which any associated *Data Element* values are specified. The unit is designated by a *unit of measure name*. When specified, the unit must be consistent with the *dimensionality* specified in the corresponding *Conceptual Domain*. Optionally, a *unit of measure precision* may be specified, as the number of decimal places to be supported in the associated *Data Element* values. This precision shall be considered a default that may be overridden for any particular *Data Element*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>unit of measure name</i>	One per <i>Unit of Measure</i>	String
<i>unit of measure precision</i>	Zero or one per <i>Unit of measure</i>	Integer

4.13 Data Element Region

The Data Element metamodel region, illustrated in Figure 9, is used to address the administration of *Data Elements*. *Data Elements* provide the formal representations for some information (such as a fact, a proposition, an observation, etc.) about some concrete or abstract thing. *Data Elements* are reusable and shareable representations of *Data Element Concepts*.

4.13.1 Metadata Objects in the Data Element Region

4.13.1.1 Data Element

A *Data Element* is considered to be a basic unit of data of interest to an organization. It is a unit of data for which the definition, identification, representation, and permissible values are specified by means of a set of attributes.

NOTE 1 In general usage, the term *data element* and *data element type* are used interchangeably. In this document, the shorter term *data element* is used.

As an *Administered Item*, a *Data Element* carries its own *Administration Record* information, allowing it to be identified, named, defined and optionally classified in a *Classification Scheme*.

A *Data Element* may be associated with other *Administered Items*, including other *Data Elements*, via the *Administered Item Relationship* shown in Figure 3 in the Administration and Identification Region. The nature of the relationship is described using the *administered item relationship type description*.

A *Data Element* is formed when a *Data Element Concept* is assigned a representation. One of the key components of a representation is the *Value Domain*, i.e., restricted valid values.

A *Data Element* is the association among a *Data Element Concept*, a *Value Domain* and optionally a *Representation Class*. The association of a *Data Element* with a *Representation Class* may be either direct (as shown by the relationship *data element representation class* in Figure 9), or via the *Value Domain* (as shown by the relationship *value domain representation class* in Figure 9). A *Data Element* cannot be registered as an *Administered Item* without being associated with a *Data Element Concept* and a *Value Domain*.

A *representation class qualifier* may be specified, that is used to qualify the name of the data element.

A *data element precision* may be used to specify the number of decimal places permitted in any associated data element values. If not specified, the *unit of measure precision* from the associated *Value Domain* shall apply.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>data element administration record</i>	One per <i>Data Element</i>	<i>Administration_Record</i>
<i>representation class qualifier</i>	Zero or one per <i>Data Element</i>	String
<i>data element precision</i>	Zero or one per <i>Data Element</i>	Integer

4.13.1.2 Data Element Concept

Data Element Concept is described under the Data Element Concept region in sub-clause 4.10. A *Data Element Concept* may be associated with several *Value Domains* resulting in a different *Data Element* for each association.

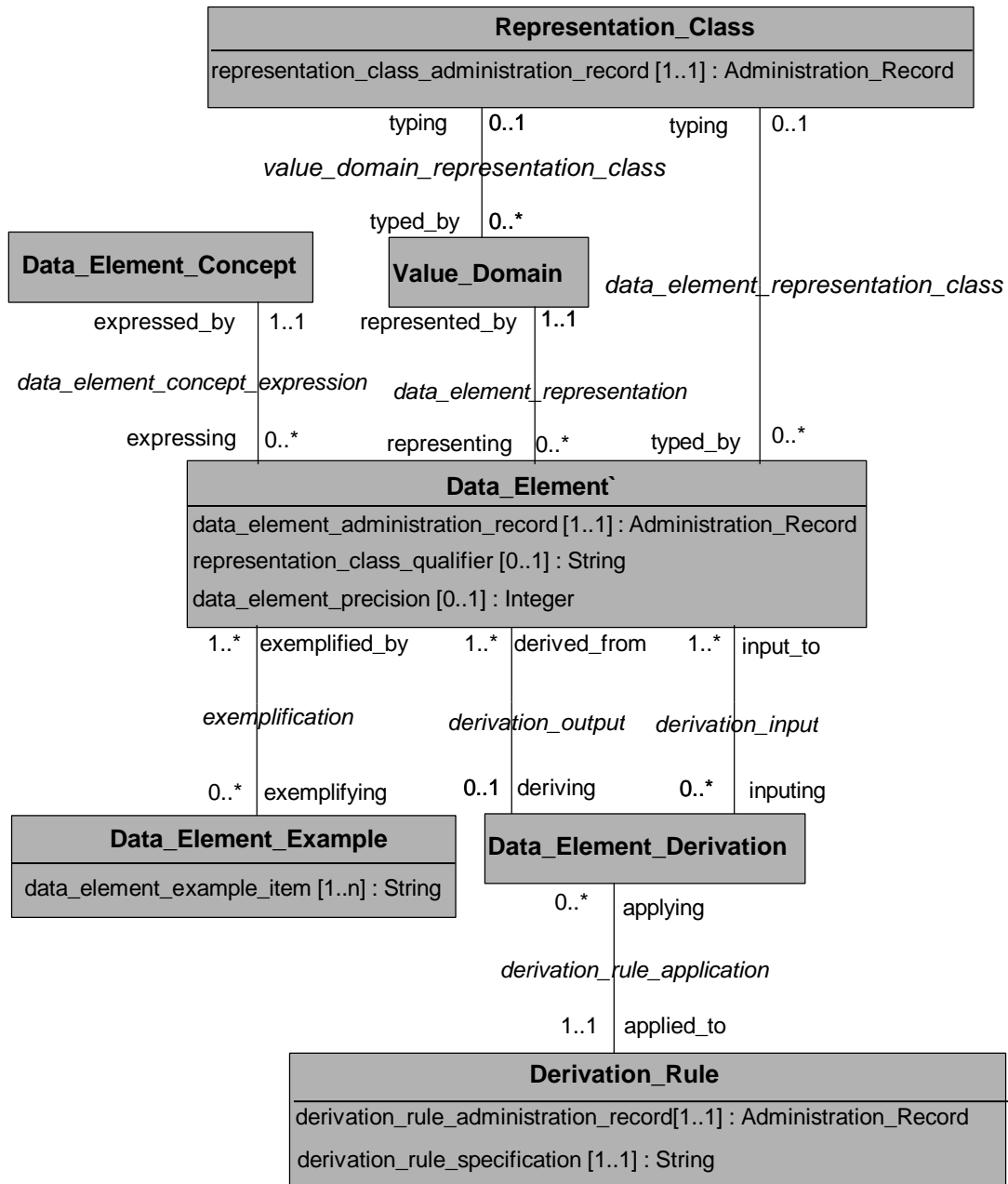


Figure 9: Data Element metamodel region

4.13.1.3 Value Domain

Value Domain is described under the Conceptual Domain and Value Domain region in sub-clause 4.11. A *Value Domain* provides representation, but has no implication as to what *Data Element Concept* the values are associated with, nor what the values mean. A *Value Domain* may be associated with multiple *Data Elements*.

4.13.1.4 Representation Class

Representation Class is the *Classification Scheme* for representation. The set of classes make it easy to distinguish among the elements in the registry. For instance, a data element categorized with the representation class 'amount' is different from an element categorized as 'number'. It probably won't make sense to compare the contents of these elements, or perform calculations using them together.

As an *Administered Item*, a *Representation Class* carries its own *Administration Record* information, allowing it to be identified, named, defined and optionally classified in a *Classification Scheme*.

A *Representation Class* may be associated with other *Administered Items*, including other *Representation Classes*, via the *Administered Item Relationship* shown in Figure 3 in the Administration and Identification Region. The nature of the relationship is described using the *administered item relationship type description*.

The major intent of *Representation class* is to provide a discrete and complete set of high-level (coarse granularity) definitions for data element/value domain categorization. This is an aid to the user in terms of application of business rules.

Representation Class is a mechanism by which the functional and/or presentational category of an item may be conveyed to a user.

An informational list of representation class terms is provided in ISO/IEC 11179-5. The list below has been expanded to provide a more comprehensive list of examples.

- Code — A system of valid symbols that substitute for specified values e.g. alpha, numeric, symbols and/or combinations.
- Count — Non-monetary numeric value arrived at by counting.
- Currency — Monetary representation
- Date — Calendar representation e.g. YYYYMMDD
- Graphic — Diagrams, graphs, mathematical curves, or the like – usually a vector image.
- Icon — A sign or representation that stands for its object by virtue of a resemblance or analogy to it
- Picture — A visual representation of a person, object, or scene – usually a raster image.
- Quantity — A continuous number such as the linear dimensions, capacity/amount (non-monetary) of an object
- Text – A text field that is usually unformatted.
- Time — Time of day or duration eg HH:MM:SS.SSSS.

None of the terms in this list is required in any specific implementation of representation class.

By using representation class, enhanced semantic control over the contents of value domains can be maintained. Rules can be drawn against representation classes that allow enforcement of content within and among value domains. For example:

- "A number-class data element cannot be used in a calculation."
- "A date-class data element must be in the format YYYYMMDD."

- "A relationship must exist between a code representation and the specific form of the value meanings which the code represents."

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>representation class</i> <i>administration record</i>	One per <i>Representation Class</i>	<i>Administration_Record</i>

4.13.1.5 Data Element Example

A *Data Element* may have *Data Element Examples* that are used to provide representative samples of the *Data Element*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>data element example item</i>	One or more per <i>Data Element Example</i>	String

4.13.1.6 Derivation Rule

A *Data Element* may have a *Derivation Rule* that is a specification of derivation for the *data element*. The *derivation rule* may range from a simple operation such as subtraction to a very complex set of derivations (derivation being defined as a relationship between a *derivation rule* and an input set upon which it acts). *Derivation rules* are not limited to arithmetic and logical operations.

As an *Administered Item*, a *Derivation Rule* carries its own *Administration Record* information, allowing it to be identified, named, defined and optionally classified in a *Classification Scheme*.

A *Derivation Rule* may be associated with other *Administered Items*, including other *Derivation Rules*, via the *Administered Item Relationship* shown in Figure 3 in the Administration and Identification Region. The nature of the relationship is described using the *administered item relationship type description*. A *Derivation Rule* may be registered as an *Administered Item* without necessarily being associated with any *Data Element Derivation*.

<u>Attribute</u>	<u>Allowed Occurrences</u>	<u>Datatype</u>
<i>derivation rule administration record</i>	One per <i>Derivation Rule</i>	<i>Administration_Record</i>
<i>derivation rule specification</i>	One per <i>Derivation Rule</i>	String

4.13.1.7 Data Element Derivation

A *Data Element Derivation* is the application of a *Derivation Rule* to one or more input *Data Elements*, to derive one or more output *Data Elements*.

4.14 Metamodel Overview

The high-level metamodel for the management of shareable data is shown in Figure 10.

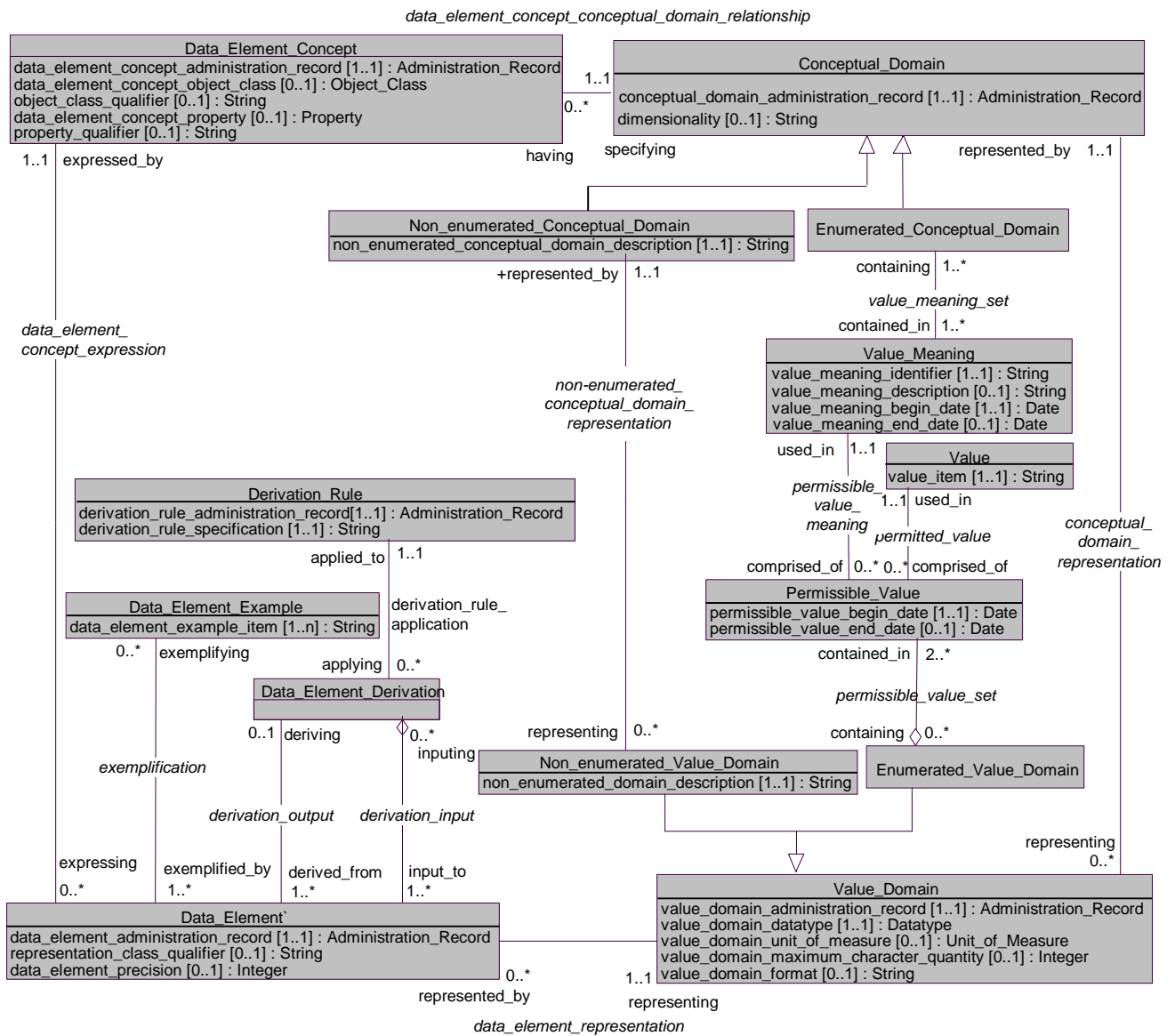


Figure 10: High-level metamodel

5 Basic Attributes of Metadata

5.1 Use of basic attributes

Clause 4 describes a model for specifying metadata in a registry. However, sometimes the requirement for metadata specification exists outside the context of a registry, for example as part of an International Standard.

A specification of metadata consists of a set of attributes, and relationships among those attributes. This clause specifies a set of *basic* attributes to be used in contexts other than a metadata registry. *Basic* means that they are frequently needed to specify a metadata item. The attributes specified in this clause are also considered *basic* in the sense that additional attributes may be required when the metadata items are used in a particular context.

Basic does not imply that all standardized attributes presented in this clause are required in all cases. Distinction is made between those basic attributes that are:

- mandatory: always required;
- conditional: required to be present under certain specified conditions;
- optional: allowed but not required.

NOTE The obligations specified for some basic attributes (especially identifiers) in contexts other than a registry are different from those specified for metadata items in a registry, as defined in clause 4.

This clause is intended to provide continuity from ISO/IEC 11179-3:1994, which edition focused on basic attributes of data elements. However, the scope of this clause extends beyond just data elements, to include: data element concepts, conceptual domains, value domains, permissible values and value meanings. A mapping among the 1994 basic attributes, the 2002 basic attributes and the 2002 metamodel can be found in Annex C.

5.2 Common attributes

The attributes listed in this sub-clause are common to all types of Administered Item. These attributes are further categorized as: Identifying, Definitional, Administrative, and Relational.

5.2.1 Identifying

<u>Attribute</u>	<u>Allowed Occurrences</u>
<i>name</i>	One or more per metadata item (<i>see note 1</i>)
<i>context name</i>	Zero or more per metadata item. Required if more than one <i>name</i> attribute exists.
<i>context identifier</i>	Zero or one per metadata item. Required if <i>context name</i> is not unique within its usage context (e.g. a standard).
<i>context description</i>	One per <i>context name</i> .
<i>item identifier</i>	Zero or one per metadata item. Required if <i>name</i> is not unique within a given <i>context</i> . (<i>see note 2</i>)
<i>item identifier – data identifier</i>	One per <i>item identifier</i> . (The mandatory portion of an <i>item identifier</i> .)
<i>item identifier – item registration authority identifier</i>	Zero or one per <i>item identifier</i> . (The optional portion of an <i>item identifier</i> - <i>see note 3</i> .)
<i>version</i>	Zero or one per metadata item. (<i>see note 4</i>)

NOTE 1 If more than one *name* is specified within a given *context*, it is usual nominate one name as "preferred", and the others as "synonyms".

NOTE 2 While *item identifier* is mandatory within a registry (see sub-clause 4.8.1.5, it is only conditional in non-registry usages. The requirement for an *item identifier* can be eliminated by qualifying *name* and/or *context name* to ensure that the combination is unique.

NOTE 3 While *item registration authority identifier* is mandatory within a registry (see sub-clause 4.8.1.5), it is optional in non-registry settings.

NOTE 4 Within a registry, *version* is part of an *item identifier*. In non-registry settings, *version* may be used independently of *item identifier*.

5.2.2 Definitional

Attribute

Allowed Occurrences

definition

One for each *context* in which the metadata item is used. (see note 1)

definition language identifier

Zero or one per *definition*.

definition source reference

Zero or one per *definition*.

NOTE Where multiple *definitions* are assigned to the same metadata item, the semantics of the *definition* should be the same across all *contexts*. (If the semantics are different, separate metadata items should be specified.) However, the terminology used to express the semantics may need to be different in different *contexts*, and thus separate *definitions* are permitted for each *context*.

5.2.3 Administrative

Administrative attributes are primarily associated with recording metadata items in a registry. They are therefore optional in non-registry settings.

Attribute

Allowed Occurrences

comments

Zero or one per metadata item.

registration status

Zero or one per metadata item.

responsible organization name

Zero or one per metadata item.

submitting organization name

Zero or one per metadata item.

5.2.4 Relational

Attribute

classification scheme name

classification scheme identifier

classification scheme type name

classification scheme item type name

classification scheme item value

related metadata reference

type of relationship

Allowed Occurrences

One for each *classification scheme* in which a metadata item is classified.

Zero or one per *classification scheme name*.
Required if *classification scheme name* is not unique within a *context*.

One for each *classification scheme* in which a metadata item is classified.

Zero or one for each *classification scheme* in which a metadata item is classified. (see note 1)

One for each *classification scheme item* by which a metadata item is classified.

Zero or more per metadata item. (see note 2)

One per *related metadata reference*.

NOTE 1 The metamodel in clause 4.10.1 treats *keywords* as a type of *classification scheme*.

NOTE 2 The metamodel allows any **Administered Item** to reference any other **Administered Item** in the same metadata registry via the **Administered Item Relationship**. For an external reference, a **Registration Authority** could choose to use a **Reference Document**, an **administrative note** or an **explanatory comment**.

5.3 Attributes specific to Data Element Concepts

The attributes listed in this sub-clause are specific to Data Element Concepts.

Attribute

object class name

object class identifier

property name

property identifier

Allowed Occurrences

One per *data element concept*.

Zero or one per *object class name*.

One per *data element concept*.

Zero or one per *property name*.

5.4 Attributes specific to Data Elements

The attributes listed in this sub-clause are specific to Data Elements.

<u>Attribute</u>	<u>Allowed Occurrences</u>
<i>Value domain name</i>	Zero or one per <i>data element</i> .
<i>Value domain identifier</i>	Zero or one per <i>data element</i> .
<i>Datatype name</i>	Zero or one per <i>data element</i> . Required if neither <i>value domain name</i> nor <i>value domain identifier</i> is not specified.
<i>Datatype scheme reference</i>	Zero or one per <i>datatype name</i> .
<i>Layout of representation</i>	Zero or one per <i>data element</i> .
<i>Representation class</i>	Zero or one per <i>data element</i> .
<i>Maximum size</i>	Zero or one per <i>data element</i> .
<i>Minimum size</i>	Zero or one per <i>data element</i> .

5.5 Attributes specific to Conceptual Domains

The attributes listed in this sub-clause are specific to Conceptual Domains.

<u>Attribute</u>	<u>Allowed Occurrences</u>
<i>dimensionality</i>	Zero or one per <i>conceptual domain</i> .

5.6 Attributes specific to Value Domains

The attributes listed in this sub-clause are specific to Value Domains.

<u>Attribute</u>	<u>Allowed Occurrences</u>
<i>datatype name</i>	One per <i>value domain</i> .
<i>datatype scheme reference</i>	Zero or one per <i>datatype name</i> .
<i>unit of measure name</i>	Zero or one per <i>value domain</i> .

5.7 Attributes specific to Permissible Values

The attributes listed in this sub-clause are specific to Permissible Values.

<u>Attribute</u>	<u>Allowed Occurrences</u>
<i>value</i>	One per <i>permissible value</i> .
<i>permissible value begin date</i>	Zero or one per <i>permissible value</i> .
<i>permissible value end date</i>	Zero or one per <i>permissible value</i> .

5.8 Attributes specific to Value Meanings

The attributes listed in this sub-clause are specific to Value Meanings.

Attribute

value meaning description

value meaning identifier

value meaning begin date

value meaning end date

Allowed Occurrences

One per *value meaning*.

Zero or one per *value meaning*.

Zero or one per *value meaning*.

Zero or one per *value meaning*.

6 Conformance

This part of ISO/IEC 11179 prescribes a conceptual model, not a physical implementation. Therefore, the metamodel need not be physically implemented exactly as specified. However, it must be possible to unambiguously map between the implementation and the metamodel in both directions.

This part of ISO/IEC 11179 also prescribes a list of basic attributes for situation where a full conceptual model is not required or not appropriate.

Conformance may be claimed to either the conceptual model, or the basic attributes or both; see 6.2. Conformance claims shall specify a Degree and a Level of Conformance, as described below.

6.1 Degree of Conformance

The distinction between "strictly conforming" and "conforming" implementations is necessary to address the simultaneous needs for interoperability and extensions. This part of ISO/IEC 11179 describes specifications that promote interoperability. Extensions are motivated by needs of users, vendors, institutions, and industries, and:

- a) are not directly specified by this part of ISO/IEC 11179,
- b) are specified and agreed to outside this part of ISO/IEC 11179, and
- c) may serve as trial usage for future editions of this part of ISO/IEC 11179.

A strictly conforming implementation may be limited in usefulness but is maximally interoperable with respect to this part of ISO/IEC 11179. A conforming implementation may be more useful, but may be less interoperable with respect to this part of ISO/IEC 11179.

6.1.1 Strictly conforming implementations

A strictly conforming implementation:

- a) shall support all mandatory, optional and conditional data element attributes and relationships;
- b) shall not use, test, access, or probe for any extension features nor extensions to data element attributes;
- c) shall not recognize, nor act on, nor allow the production of data element attributes that are dependent on any unspecified, undefined, or implementation-defined behavior.

NOTE The use of extensions to the metamodel or the basic attributes may cause undefined behavior.

6.1.2 Conforming implementations

A conforming implementation:

- a) shall support all mandatory, optional and conditional data element attributes and relationships;
- b) as permitted by the implementation, may use, test, access, or probe for extension features or extensions to data element attributes;
- c) may recognize, act on, or allow the production of data element attributes that are dependent on implementation-defined behavior.

NOTE 1 All strictly conforming implementations are also conforming implementations.

NOTE 2 The use of extensions to the metamodel or the basic attributes may cause undefined behavior.

6.2 Levels of Conformance

An implementation may conform to either of two levels of conformance to this standard:

6.2.1 Conformance Level 1

Only those metadata elements, relationships and properties specified in Clause 5 are supported and used;

6.2.2 Conformance Level 2

All metadata elements, relationships and properties specified in Clause 4 are supported and may be used.

6.3 Obligation

Properties and relationships specified in this part of ISO/IEC 11179 are stated to be Mandatory, Conditional or Optional.

For the purpose of conformance:

- a) Mandatory properties and relationships shall exist, and shall conform to the provisions of this part of ISO/IEC 11179.
- b) Anything specified as Conditional within this part of ISO/IEC 11179 shall be treated as Mandatory if the associated condition is satisfied, and shall otherwise be not present.
- c) Optional properties and relationships are not required to exist, but if they do exist they shall conform to the provisions of this part of ISO/IEC 11179.

Such obligation is enforced if and only if the Registration Status of the associated metadata items is Recorded or higher.

6.4 Conformance to prior editions of this Standard

The following are the registry items and their obligation attributes in ISO/IEC 11179-3:1994, and whether each item is now considered obsolete:

- **Identifying:** Name (mandatory), Identifier (conditional), Version (conditional), Registration Authority (conditional), Synonymous Name (optional, *obsolete*), Context (conditional)
- **Definitional:** Definition (mandatory)
- **Relational:** Classification Scheme (optional), Keywords (optional, *obsolete*), Related Data Reference (optional, *obsolete*), Type of Relationship (conditional)
- **Representational:** Representation Category (mandatory, *obsolete*), Form of Representation (mandatory, *obsolete*), Datatype of Data Element Values (mandatory), Maximum Size of Data Element Values (mandatory), Minimum Size of Data Element Values (mandatory, *obsolete*), Layout of Representation (conditional, *obsolete*), Permissible Data Element Values (mandatory, *obsolete*)
- **Administrative:** Responsible Organization (optional), Registration Status (conditional), Submitting Organization (optional), Comments (optional)

Annex C relates the attributes of the 1994 attributes to the new metamodel.

6.5 Implementation Conformance Statement (ICS)

An implementation claiming conformance to this part of ISO/IEC 11179 shall include an Implementation Conformance Statement stating:

- d) whether it conforms or strictly conforms (6.1);
- e) whether conformance is to Level 1, Level 2 (6.2) or both;
- f) what extensions are supported or used.

6.6 Roles and Responsibilities for Registration

Conformance needs to be considered in the context of the roles and responsibilities of registration authorities, as covered by ISO/IEC 11179-6: Registration of data elements.

Extended conformance of systems requires formalisation of procedures, agreement of roles and responsibilities between parties, and guidelines addressing use of software products and conversions from other systems. The formalisation of these aspects must be consistent with the conformance requirements in the above clauses, and roles of registration authorities as set out in ISO/IEC 11179-6.

Annex A (Informative) Alphabetical List of Terms

EDITOR'S NOTE: The clause numbers need to be updated.

Term	Defined in
Administered Item	3.3.1
administered item classification	3.3.2
administered item context	3.3.3
administered item identifier	3.3.4
Administered Item Relationship	3.3.5
administered item relationship type description	3.3.6
Administration Record	3.3.7
administrative note	3.3.8
administrative status	3.3.9
association	3.1.1
association class	3.1.2
attribute	3.1.3
attribute capsule	3.1.4
attribute value	3.2.1
basic attribute	3.2.2
binding	3.2.3
change description	3.3.10
characteristic	3.2.4
class	3.1.5
Classification Scheme	3.3.11
classification scheme administration record	3.3.12
Classification Scheme Item (CSI)	3.3.13
Classification Scheme Item Relationship	3.3.14
classification scheme item relationship type description	3.3.15
classification scheme item type name	3.3.16
classification scheme item value	3.3.17
classification scheme membership	3.3.18
classification scheme type name	3.3.19
common attribute	3.2.5

Term	Defined in
Concept	3.3.20
Concept Relationship	3.3.21
concept relationship type description	3.3.22
conceptual data model	3.2.6
Conceptual Domain (CD)	3.3.23
conceptual domain administration record	3.3.24
Conceptual Domain Representation	3.3.25
conditional	3.2.7
Contact	3.3.26
contact information	3.3.27
contact name	3.3.28
contact title	3.3.29
Context (for administered item)	3.3.30
context administration record	3.3.31
context description	3.3.32
context description language identifier	3.3.33
country identifier (of Language Identification)	3.3.34
creation date	3.3.35
data	3.2.8
Data Element (DE)	3.3.36
data element administration record	3.3.37
Data Element Concept (DEC)	3.3.38
data element concept administration record	3.3.39
data element concept conceptual domain relationship	3.3.40
data element concept expression	3.3.41
data element concept object class	3.3.42
data element concept property	3.3.43
Data Element Derivation	3.3.44
Data Element Example	3.3.45

Term	Defined in
data element example item	3.3.46
data element precision	3.3.47
data element representation	3.3.48
data element representation class	3.3.49
data identifier	3.3.50
data model	3.2.9
Datatype	3.3.51
datatype annotation	3.3.52
datatype description	3.3.53
datatype name	3.3.54
datatype scheme reference	3.3.55
Definition (of Administered Item)	3.3.56
definition source reference	3.3.57
definition text	3.3.58
derivation input	3.3.59
derivation output	3.3.60
Derivation Rule	3.3.61
derivation rule administration record	3.3.62
derivation rule application	3.3.63
derivation rule specification	3.3.64
designation	3.2.10
Designation (of Administered Item)	3.3.65
dimensionality (of Conceptual Domain)	3.3.66
documentation language identifier	3.3.67
effective date	3.3.68
encapsulated class	3.1.6
entity	3.2.11
Enumerated Conceptual Domain	3.3.69
Enumerated Value Domain	3.3.70
exemplification	3.3.71
explanatory comment	3.3.72
extension	3.2.12
extension (of registry metamodel)	3.2.13
generalization	3.1.7
identifier (in Metadata Registry)	3.1.8
International code designator	3.3.73
Item identifier	3.3.74
item registration authority identifier	3.3.75
language	3.2.14

Term	Defined in
Language Identification	3.3.76
language identifier	3.3.77
Language Section	3.3.78
language section language identifier	3.3.79
last change date	3.3.80
mandatory	3.2.15
metadata	3.2.16
metadata item	3.2.17
metadata object	3.2.18
metadata register	3.2.19
Metadata Registry (MdR)	3.2.20
metadata set	3.2.21
metamodel	3.2.22
metamodel construct	3.2.23
name	3.2.24
name (of Administered Item)	3.3.81
Non-enumerated Conceptual Domain	3.3.82
non-enumerated conceptual domain description	3.3.83
Non-enumerated Value Domain	3.3.84
non-enumerated value domain description	3.3.85
object	3.2.25
Object Class	3.3.86
object class administration record	3.3.87
object class qualifier	3.3.88
optional	3.2.26
Organization	3.3.89
organization identifier	3.3.90
organization mail address	3.3.91
organization name	3.3.92
organization part	3.3.93
organization part identifier (opi)	3.3.94
organization part identifier source	3.3.95
origin (of Administered Item)	3.3.96
Permissible Value	3.3.97
permissible value begin date	3.3.98
permissible value end date	3.3.99
permissible value meaning	3.3.100
permissible value set	3.3.101

Term	Defined in
permitted value	3.3.102
preferred definition	3.3.103
preferred designation	3.3.104
Property	3.3.105
property administration record	3.3.106
property qualifier	3.3.107
reference	3.3.108
Reference Document	3.3.109
reference document identifier	3.3.110
reference document language	3.3.111
reference document title	3.3.112
reference document type description	3.3.113
reference organization	3.3.114
Registrar	3.3.115
registrar identifier	3.3.116
registration	3.3.117
Registration Authority	3.3.118
registration authority identifier	3.3.119
Registration Authority Identifier	3.3.120
registration authority registrar	3.3.121
registration status	3.3.122
registry item	3.2.27
registry metamodel	3.2.28
related metadata reference	3.2.29
relationship (in registry metamodel)	3.1.9
Representation Class	3.3.123
representation class administration record	3.3.124
representation class qualifier	3.3.125

Term	Defined in
Stewardship (of Administered Item)	3.3.126
stewardship (of metadata)	3.2.30
stewardship contact	3.3.127
Submission (of Administered Item)	3.3.128
submission contact	3.3.129
Terminological Entry	3.3.130
Unit of Measure (of Value Domain)	3.3.131
unit of measure name	3.3.132
unit of measure precision	3.3.133
unresolved issue	3.3.134
until date	3.3.135
Value	3.3.136
Value Domain	3.3.137
value domain administration record	3.3.138
value domain datatype	3.3.139
value domain format	3.3.140
value domain maximum character quantity	3.3.141
value domain representation class	3.3.142
value domain unit of measure	3.3.143
value item	3.3.144
Value Meaning	3.3.145
value meaning begin date	3.3.146
value meaning description	3.3.147
value meaning end date	3.3.148
value meaning identifier	3.3.149
value meaning set	3.3.150
version	3.3.151

Annex B (Informative) Modelling Notation

NOTE In this part of ISO/IEC 11179, the metamodel in clause 4 is expressed using the Unified Modeling Language (UML). This annex introduces the UML syntax.

B.1 Modelling symbols

The Unified Modeling Language (UML) is used to describe this metamodel. This notation is particularly suited to documenting a conceptual data schema. The structure used in the description (the metamodel) is compatible and is partially described by the metamodel described. Since this is a *conceptual data model*, only the *data element concept* information is used. A more complete description of UML may be found in ISO/IEC 19501-1:2001, Information Technology – Unified Modeling Language – Part 1: Specification.

The object model provides for four basic types of Modeling objects: *classes* (shown as rectangles in a diagram), *associations* among these classes (shown as lines), *operations*, and *attributes* that are associated with these classes. For example, the attributes describing employees and the cars they own could be modelled as follows:

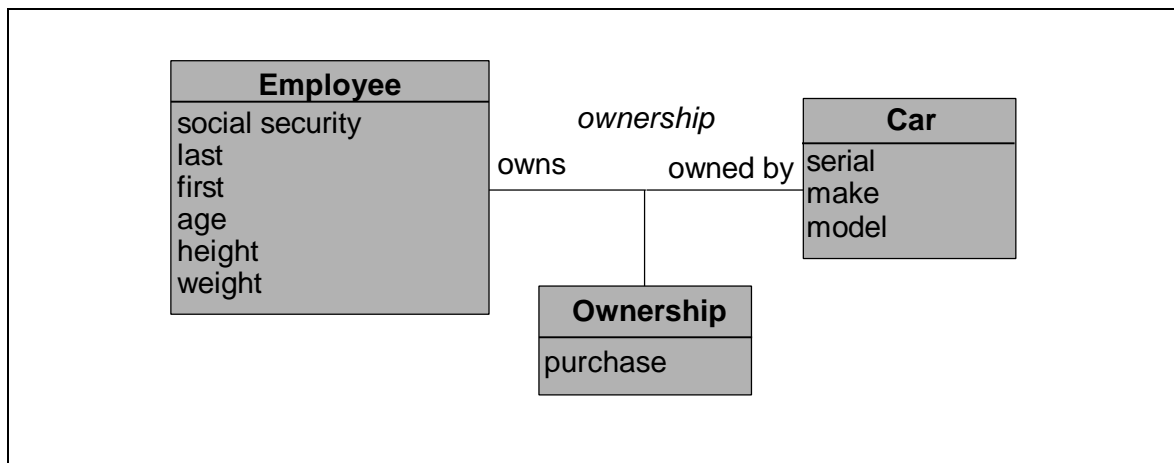


Figure B-1: Sample modelling diagram

At the current time operations are not shown.

B.1.1 Classes

Classes (Entities) are represented by rectangles and are the things about which the business processes information. A class is something that has distinctiveness and typically, properties of its own. Classes may be persons, places, concepts, events, or other fundamental things. For example, employee and car are both classes. The name of the class appears in the top area of the class rectangle.

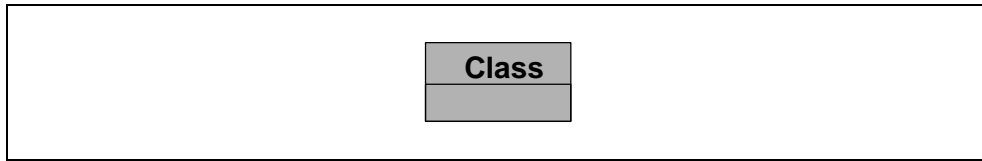


Figure B-2: Notation for "Class"

As described below, entities may have attributes.

B.1.2 Associations

Classes have business-based *associations (relationships)* with one or more other classes. These associations are bi-directional. For example, an employee owns a car. Conversely the car is owned by the employee. This association between car and employee may be called ownership. In the UML diagram these links between classes are represented by lines.

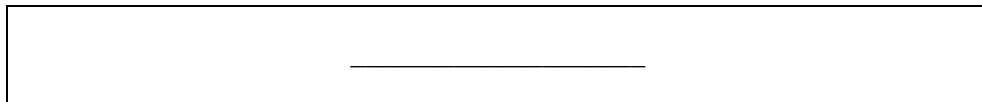


Figure B-3: Notation for "Association"

Lines are drawn between the rectangles to identify the classes participating in an association. The role of each class in the association is specified at the corresponding end of the association line. The name of the *association* is shown near the centre of the association line.

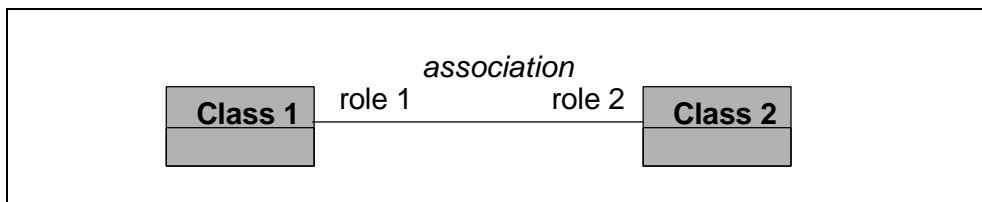


Figure B-4: Notation for relationship between Classes

The statement of the number of class instances that may participate at each end of an association is its *cardinality*. It is expressed as a minimum and a maximum, separated by a "..".



Figure B-5: Notation for relationship with Cardinality

This example states that class 1 may be associated with a minimum of one and a maximum of many occurrences of class 2. Class 2 may be associated with a minimum of zero (i.e., the association is optional) and a maximum of one occurrence of class 1.

As described below, like classes, association classes may also have attributes.

B.1.3 Association class

There are situations where a particular business object fits the criteria for both a class and an association. This usually occurs when an association itself is found to have a business association with a class. Thus, an *association class* is defined as an association that acts like a class. It is represented as a class that is connected to an association with a dashed line.

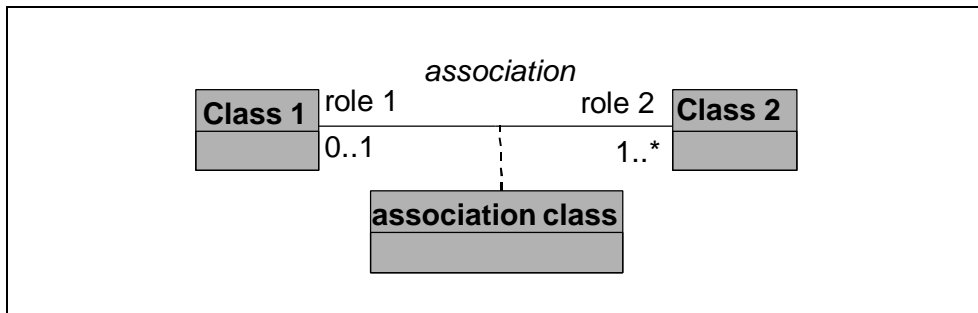


Figure B-6: Notation for "Association Class"

B.1.4 Subtypes

Classes may be decomposed into a hierarchy with increasing level of details. Each *supertype* class may have *subtypes*. This is a generalization/specialization association.

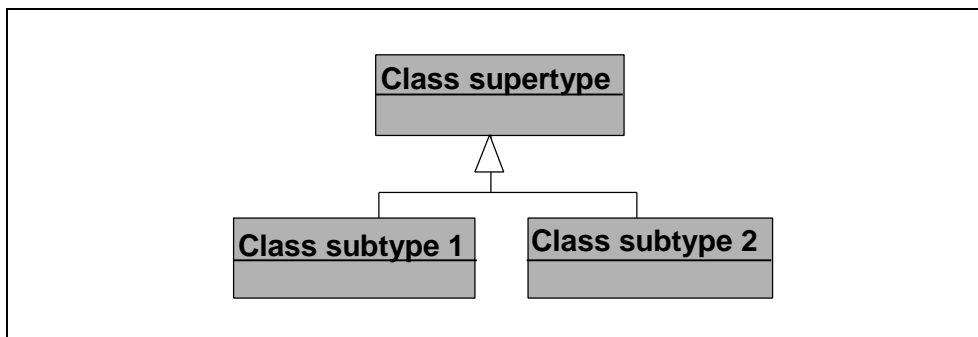


Figure B-7: Notation for Supertype / Subtypes

Subtypes inherit all the attributes of their supertype. In addition to the attributes that are inherited, each subtype may have unique attributes of its own. In addition, subtypes participate in all the associations in which their supertype participates.

B.1.5 Aggregation

A class sometimes consists of component classes. When a class consists of assemblies of "parts", there is a special relationship between the component parts and the class representing the assembly of these parts. An aggregation is represented by a hollow diamond at the end of the association line. The tip of the diamond points toward the assembly class.

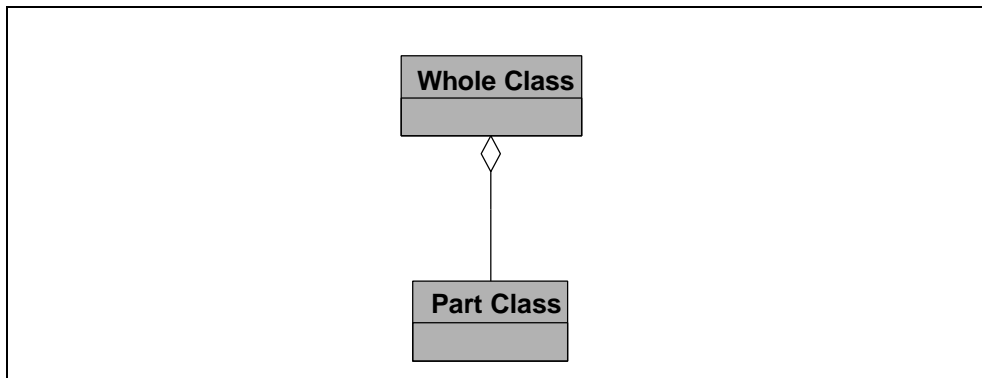


Figure B-8: Notation for Aggregation

B.1.6 Composite Aggregation

A class sometimes consists of component classes where the part classes cannot exist without the whole class. When a class consists of assemblies of “parts”, there is a special relationship between the component parts and the class representing the assembly of these parts. A composite aggregation is represented by a solid diamond at the end of the association line. The tip of the diamond points toward the assembly class.

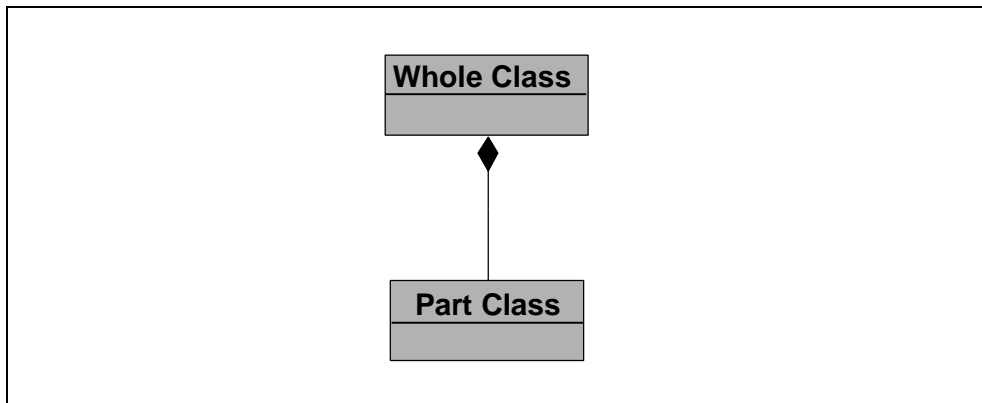


Figure B-9: Notation for Composite Aggregation

B.1.7 Attribute

The *properties* (or characteristics) of a class or an association class are described as *attributes*. Each attribute represents one *fact*. Alone among the other information modeling objects, attributes have actual *permissible values*. In this document, we display attribute names in the area beneath the class name. The obligation of the attribute is indicated using [1..1] for required attributes, and [0..1] for optional attributes. An attribute may have a datatype that is shown separated from the attribute name (and optional obligation) by a colon.

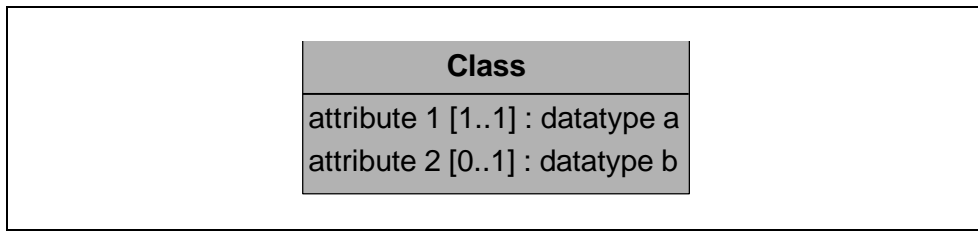


Figure B-10: Notation for Class with Attributes

B.1.8 Attribute Capsules and Encapsulated classes

A datatype may itself be a class. In this document, the term *attribute capsule* is used to refer to an attribute that encapsulates a class in its datatype. The term *encapsulated class* is used to refer to the class that is encapsulated..

NOTE Neither *attribute capsule* nor *encapsulated class* is specified as a term in ISO/IEC 19501-1.

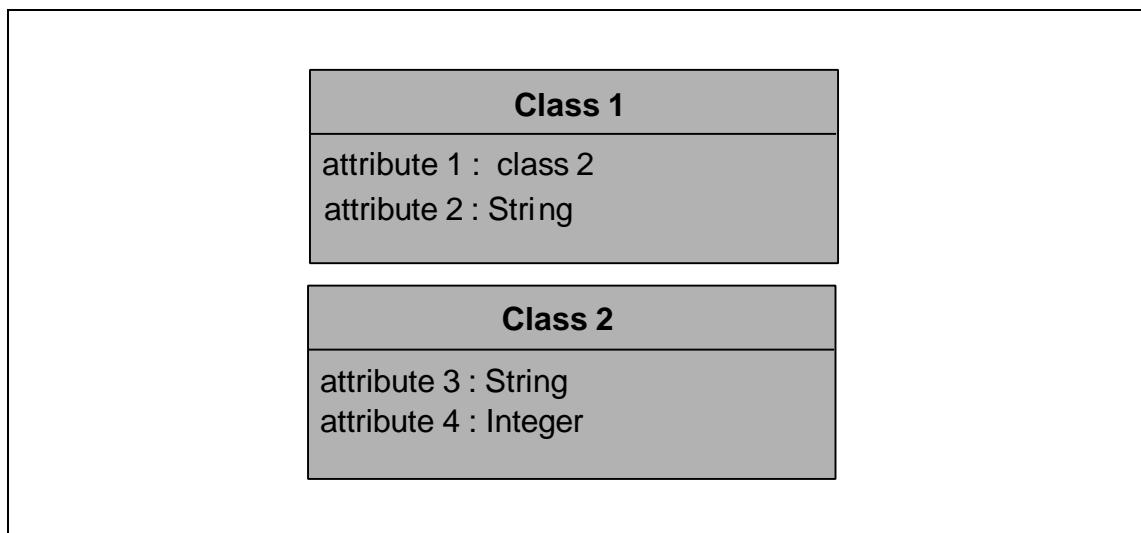


Figure B-11: Notation for Attribute Capsules

Annex C (Informative) Mapping the ISO/IEC 11179-3:1994 basic attributes to the ISO/IEC 11179-3:2002 metamodel and basic attributes

C.1 Introduction

ISO/IEC 11179-3:1994 lists 23 basic attributes of data elements, as shown in Figure C-1.

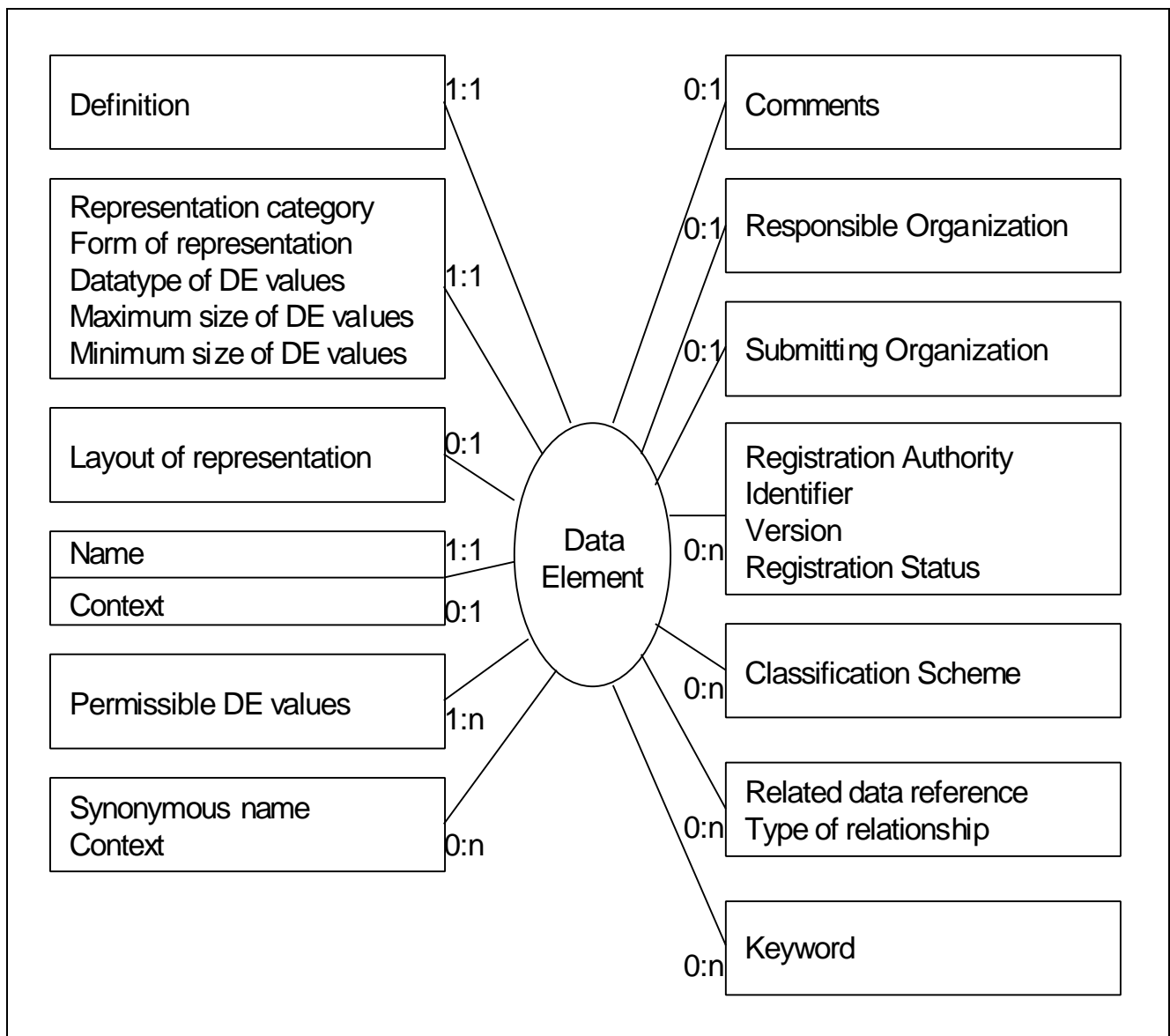


Figure C-1: Basic Attributes of Data elements

This edition of the standard supports not only data elements, but also other metadata items associated with them, such as data element concepts, conceptual domains and value domains.

This annex maps the 1994 basic attributes to the metamodel in clause 4, and the new basic attributes in clause 5.

C.1.1 Description of Table Structures in this Annex

The tables in this Annex are structured as follows:

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:			
Path from Administered Item			
Definition:			
Obligation:			
Condition:			
Datatype:			
Comment:			

C.1.1.1 Description of the Columns

The columns in the table are used as follows:

- Column 1: Label for the row
- Column 2: What was specified in ISO/IEC 11179-3:1994 clause 6
- Column 3: What is specified in ISO/IEC 11179-3:2002 clause 4
- Column 4: What is specified in ISO/IEC 11179-3:2002 clause 5

C.1.1.2 Description of the Rows

The rows in the table are used as follows, with the value in a particular cell coming from the clause identified by the column (see above).

- Row 1: Attribute name - Contains the name of the attribute. For column 3, this is specified as: "Class name" . "attribute name", where "Class name" designates the Class in the metamodel that contains the attribute.
- Row 2: Path from Administered Item – Used for column 3 only. Specifies the path that needs to be navigated in the metamodel to reach this attribute for any particular Administered Item. (See next sub-clause for an explanation of the notation.) Whenever the attribute is on the Administered Item class, no navigation is necessary and this row is omitted.
- Row 3: Definition – Contains the definition of the attribute.
- Row 4: Obligation – Contains the obligation of the attribute. (One of: Mandatory, Optional or Conditional.)
- Row 5: Condition – If the Obligation is "Conditional", this row contains the condition that applies. (The entire row is omitted if it is not relevant for any column.)
- Row 6: Datatype – Contains the datatype of the attribute.

- Row 7: Comment – Contains any explanatory comment. (The entire row is omitted if it is not relevant for any column.)

The attributes are ordered in this Annex as in clause 5 of this document.

The notation "N/A" indicates that a row is "Not Applicable" for a particular column.

C.1.1.3 Specification of attribute name in row 1 column 3

For the old and new basic attributes (columns 2 and 4 respectively) the attribute name is straightforward. The equivalent attributes in the metamodel (column 3), need to be designated in the context of a particular class. The class that provides the context is named first, and then the attribute, using the "dot" notation:

"Class Name" . "attribute name"

e.g. "Item Identifier" . "version"

C.1.1.4 Specification of path from Administered Item in row 2 column 3

In addition to designating the metamodel attribute in the context of a class (row 1 column 3), row 2 column 3 shows how the class is related an Administered Item. It is necessary to navigate relationships and/or attribute capsules within the model from one class to another. For common attributes (i.e. those that apply to any Administered Item), the starting point for navigation is the supertype class "Administered Item". For attributes specific to a particular sub-type of Administered Item, the starting point for navigation is that sub-type class (e.g. Data Element). The "dot" notation is used as described below.

NOTE 1 The following notational convention is used:

- the names of classes and encapsulated classes are capitalized e.g. "Item Identifier"
- the names of attributes are all lower case e.g. "version"
- the names of relationships are lower case and italicised e.g. "*name entry*"

NOTE 2 The use of *italics* to indicate a relationship applies only to the specification of the navigation path in row 2 column 3. In row 3 Definition, *italics* are used to distinguish the term from the definition.

Example 1: Attribute "version"

In this example, the attribute is a Common Attribute (i.e. it can apply to any type of Administered Item), so the navigation starts from the super-type class "Administered Item".

"Administered Item" . "administered item administration record" .

"Administration Record" . "administered item identifier" .

"Item Identifier" . "version"

specifies to follow the path in the model:

- from the class "Administered Item" via its attribute "administered item administration record" to the encapsulated class "Administration Record", then
- from the class "Administration Record" via its attribute "administered item identifier" to the encapsulated class "Item Identifier" and its attribute "version".

Example 2: Attribute "datatype name"

In this example, the attribute is specific to a Data Element, so the navigation starts from the "Data Element" sub-type class of Administered Item.

"Data Element" . "data element representation" .
 "Value Domain" . "value domain datatype" .
 "Datatype" . "datatype name"

specifies to follow the path in the model:

- from the class "Data Element" via its relationship "data element representation" to the related class "Value Domain", then
- from the class "Value Domain" via its attribute "value domain datatype" to the encapsulated class "Datatype" and its attribute "datatype name".

C.2 Mapping the Basic Attributes

The attributes are ordered in this Annex as in clause 5 of this document.

C.2.1 Common Identifying attributes

C.2.1.1 Name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Name	"Designation (of Administered Item)" . "name"	name
Path from Administered Item:	N/A	"Administered Item" . "administered item context" "Terminological Entry" . "terminological entry languages" . "Language Section" . "name entry" .	N/A
Definition:	Single or multi word designation assigned to a data element.	A name by which an Administered Item is known within a specific Context.	A name by which a metadata item is known within a specific context.
Obligation:	Mandatory	Mandatory	Mandatory
Data type:	Character string	String	String
Comment:		The attribute "preferred designation" may be used to specify the primary name if synonyms also exist in a particular context.	

C.2.1.2 Synonymous name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Synonymous name	"Designation (of ...	name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
		Administered Item)" . "name"	
Path from Administered Item:	N/A	"Administered Item" . "administered item context" "Terminological Entry" . "terminological entry languages" . "Language Section" . "name entry" .	N/A
Definition:	Single word or multi word designation that differs from the given name, but represents the same data element concept.	A name by which an Administered Item is known within a specific Context.	A name by which a metadata item is known within a specific context.
Obligation:	Optional	Optional	Optional
Data type:	Character string	String	String
Comment:	Synonymous names are often familiar names in a certain application environment. If this is the case use attribute 'Context' (6.1.6) to specify the context. If more synonymous names occur the attributes 'Synonymous name' and 'Context' shall be specified as a pair.	An Administered Item may have multiple names in the same or different contexts. The distinction between "name" and "synonymous name" in a particular context may be specified by the attribute "preferred designation", which should be set to "True" for the preferred name, and "False" for all synonyms.	A metadata item may have multiple names in the same or different contexts. The distinction between "name" and "synonymous name" in a particular context may be specified by the attribute "preferred designation", which should be set to "True" for the preferred name, and "False" for all synonyms.

C.2.1.3 Context name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Context	"Designation (of Administered Item)" . "name" NOTE The "Administered Item" referred to here is the Context itself, not the Administered Item to which context is being provided.	context name
Path from Administered Item:	N/A	"Administered Item" ⁽¹⁾ . "administered item context" "Context" . "context administration record" . "Administration Record" . "Administered Item" ⁽²⁾ . "administered item context" "Terminological Entry" . "terminological entry languages" . "Language	N/A

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
		Section” . “name entry” . NOTES ^{(1) (2)} The first "Administered Item" is the one to which context is being provided. The second "Administered Item" is the Context itself.	
Definition:	A designation or description of the application environment or discipline in which a name and/or synonymous name is applied or originates from. Note: The latest edition of the standard differentiates designations from descriptions.	<i>Context</i> : A universe of discourse in which a name or definition is used. <i>name</i> : A name by which an Administered Item (in this case the Context) is known within a specific context (where the context for a Context is probably the registry).	<i>Context</i> : A universe of discourse in which a name or definition is used. <i>name</i> : A name by which a metadata item (in this case the Context) is known within a specific context (where the context for a context is the setting in which it is used).
Obligation:	Conditional	Mandatory	Conditional
Condition:	This attribute is mandatory for each occurrence of the attribute 'Synonymous name' (6.1.5). This attribute is mandatory when the attribute 'Name' (6.1.1) occurs in an information exchange.	N/A	Required if more than one <i>name</i> attribute exists for a particular metadata item.
Data type:	Character string	String	String
Comment:	Assignment of the attribute 'Context' to the attribute 'Name' may be made mandatory as part of the procedures of any Registration Authority.	As an Administered Item itself, any Context used within a registry must be given both a <i>name</i> and <i>definition</i> . A Context must itself exist within a Context, which for most will probably be the registry. (A Context may provide Context to itself.)	

C.2.1.4 Context identifier

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	“Context” . “context administration record” . “Administration Record” . “administered item identifier”	context identifier
Path from Administered Item:	N/A	“Administered Item” . “ <i>administered item context</i> ” .	N/A

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Definition:	N/A	<i>Context</i> : A universe of discourse in which a name or definition is used. <i>administered item identifier</i> : The unique identifier for an Administered Item (in this case the Context).	<i>Context</i> : A universe of discourse in which a name or definition is used. <i>context identifier</i> : A unique identifier for the <i>Context</i> within its usage context.
Obligation:	N/A	Mandatory	Conditional
Condition:	N/A	N/A	Required if <i>context name</i> is not unique with its usage context.
Data type:	N/A	String	String
Comment:		As an Administered Item itself, any Context used within a registry must be given an <i>administered item identifier</i> .	

C.2.1.5 Context description

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Context	"Context" . "context description"	context description
Path from Administered Item:	N/A	"Administered Item" . "administered item context"	N/A
Definition:	A designation or description of the application environment or discipline in which a name and/or synonymous name is applied or originates from. Note: The new metamodel differentiates designations from descriptions.	<i>Context</i> : A universe of discourse in which a name or definition is used. <i>context description</i> : The textual description of the context.	<i>Context</i> : A universe of discourse in which a name or definition is used. <i>context description</i> : The textual description of the context.
Obligation:	Conditional	Mandatory	Conditional
Condition:	This attribute is mandatory for each occurrence of the attribute 'Synonymous name'. This attribute is mandatory when the attribute 'Name' occurs in an information exchange.	N/A	Required if <i>context name</i> is used.
Data type:	Character string	String	String
Comment:	Assignment of the attribute	In this edition of this part of	In this edition of this part of

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
	'Context' to the attribute 'Name' may be made mandatory as part of the procedures of any Registration Authority.	ISO/IEC 11179, <i>context description</i> and <i>context name</i> exist as two separate attributes.	ISO/IEC 11179, <i>context description</i> and <i>context name</i> exist as two separate attributes.

C.2.1.6 Item identifier – data identifier

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Identifier	"Item Identifier" . "data identifier"	item identifier – data identifier
Path from Administered Item:	N/A	"Administered Item" . "administered item administration record" . "Administration Record" . "administered item identifier" .	N/A
Definition:	A language independent unique identifier of a data element within a Registration Authority.	The unique identifier for an Administered Item within a Registration Authority.	The unique identifier for a metadata item within a specific context.
Obligation:	Conditional	Mandatory	Conditional
Condition:	If the attribute 'Name of data element' is not unique within a Registration Authority this attribute is mandatory.	N/A	If the attribute <i>name</i> is not unique within a <i>context</i> , this attribute is mandatory.
Data type:	Character	String	String
Comment:	Assignment of a unique identifier may be made mandatory as part of the registration procedure of any Registration Authority.		The requirement for an <i>item identifier</i> can be eliminated by qualifying <i>name</i> and/or <i>context name</i> to ensure that the combination is unique.

C.2.1.7 Item registration authority identifier

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Registration Authority	"Item Identifier" . "item registration authority identifier"	item identifier – item registration authority identifier
Path from Administered Item:	N/A	"Administered Item" . "administered item administration record" . "Administration Record" . "administered item identifier" .	N/A

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Definition:	Any organization authorized to register data elements.	An identifier (described in ISO/IEC 11179 Part 6) assigned to the Registration Authority registering the item.	An identifier (described in ISO/IEC 11179 Part 6) assigned to the registration authority registering the item.
Obligation:	Conditional	Mandatory	Conditional
Condition:	One Registration Authority shall be specified for each Identifier present.	N/A	Required if <i>item identifier – data identifier</i> is not unique within the usage context.
Data type:	Character string	String	String

C.2.1.8 Version

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Version	"Item identifier" . "version"	Version
Path from Administered Item:	N/A	"Administered Item" . "administered item administration record" . "Administration Record" . "administered item identifier" .	N/A
Definition:	Identification of an issue of a data element specification in a series of evolving data element specifications within a Registration Authority.	The unique version identifier of the Administered Item.	The unique version identifier of the metadata item.
Obligation:	Conditional	Mandatory	Optional
Condition:	This attribute is mandatory if updates on attributes occur which meet the maintenance rules for allocating new versions as set by the Registration	N/A	N/A
Data type:	Character	String	String

C.2.2 Common Definitional attributes

C.2.2.1 Definition

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Definition	"Definition (of Administered Item)" . "definition text"	definition

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Path from Administered Item:	N/A	"Administered Item" . "administered item context" . "Terminological Entry" . "terminological entry languages" . "Language Section" . "definition entry" .	N/A
Definition:	Statement that expresses the essential nature of a data element and permits its differentiation from all other data elements.	<i>Definition:</i> The definition of an Administered Item within a Context. <i>Definition text:</i> The text of the definition.	The definition of an metadata item within a context.
Obligation:	Mandatory	Mandatory	Mandatory
Data type:	Character string	String	String
Comment:		Where more than one Definition is provided within a particular context, one of them may be specified as preferred by setting the attribute "preferred definition" to "True".	

C.2.2.2 Definition language

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Language Section" . "language section language identifier"	definition language identifier
Path from Administered Item:	N/A	"Administered Item" . "administered item context" . "Terminological Entry" .	N/A
Definition:	N/A	The identifier of the language used within the <i>Terminological Entry</i> , which applies to both the name and the definition.	The identifier of the language used within the definition.
Obligation:	N/A	Optional	Optional
Data type:	N/A	String	String

C.2.2.3 Definition source reference

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Definition" . "definition source reference"	definition source reference

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Path from Administered Item:	N/A	"Administered Item" . " <i>administered item context</i> " . "Terminological Entry" . " <i>terminological entry languages</i> " . "Language Section" . " <i>definition entry</i> " .	N/A
Definition:	N/A	A reference to the source from which the definition is taken.	A reference to the source from which the definition is taken.
Obligation:	N/A	Optional	Optional
Data type:	N/A	String	String

C.2.3 Common Administrative attributes

C.2.3.1 Comments

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Comments	"Administration Record" . "explanatory comment"	Comments
Path from Administered Item:	N/A	"Administered Item" . "administered item administration record" .	N/A
Definition:	Remarks on the data element.	Descriptive comments about the Administered Item.	Descriptive comments about the metadata item.
Obligation:	Optional	Optional	Optional
Data type:	Character string	String	String

C.2.3.2 Registration status

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Registration status	"Administration Record" . "registration status"	registration status
Path from Administered Item:	N/A	"Administered Item" . "administered item administration record" .	N/A
Definition:	A designation of the position in the registration life-cycle of a data element.	A designation of the status in the registration life-cycle of an Administered Item.	A designation of the status in the registration life-cycle of a metadata item.
Obligation:	Conditional	Mandatory	Optional
Condition:	This attribute is mandatory during the data element life-cycle specified by any Registration Authority.	N/A	N/A

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Data type:	Character	String	String
Comment:	The type of registration status to be distinguished and the allocation of the registration status shall follow the rules that are described in the procedures for the registration of data elements (see Part 6 of this International Standard).		

C.2.3.3 Responsible organization

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Responsible organization	"Organization" . "name"	Responsible organization
Path from Administered Item:	N/A	"Administered Item" . "stewardship" .	N/A
Definition:	The organization or unit within an organization that is responsible for the contents of the mandatory attributes by which the data element is specified.	<i>Organization</i> : A unique framework of authority, within which a person or persons act, or are designated to act, towards some purpose. <i>stewardship</i> : The relationship of an Administered Item, a Contact and an Organization involved in the stewardship of the metadata.	The organization or unit within an organization that is responsible for the contents of the mandatory attributes by which the metadata item is specified.
Obligation:	Optional	Mandatory	Optional
Data type:	Character string	String	String
Comment:	The organization shall be considered as 'owner' of the data element.		

C.2.3.4 Submitting organization

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Submitting organization	"Organization" . "name"	Submitting organization
Path from Administered Item:	N/A	"Administered Item" . "submission" .	N/A
Definition:	The organization or unit within an organization that	<i>Organization</i> : A unique framework of authority,	The organization or unit within an organization that

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
	has submitted the data element for addition, change or cancellation/withdrawal in the data element dictionary.	within which a person or persons act, or are designated to act, towards some purpose. <i>submission</i> : The relationship of an Administered Item, a Contact and an Organization involved in a submission of metadata.	has submitted the metadata item for addition, change or cancellation/withdrawal in a metadata registry.
Obligation:	Optional	Mandatory	Optional
Data type:	Character string	String	String

C.2.4 Common Relational attributes

C.2.4.1 Classification scheme name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Classification scheme	“Designation (of Administered Item)” . “name” NOTE The "Administered Item" referred to here is the Classification Scheme, not the Administered Item which is being classified.	Classification scheme name
Path from Administered Item:	N/A	“Administered Item” ⁽¹⁾ . “ <i>administered item classification</i> ” . “Classification Scheme Item” . “ <i>classification scheme membership</i> ” . “Classification Scheme” . “classification scheme administration record” . “Administration Record” . “Administered Item” ⁽²⁾ . “ <i>administered item context</i> ” . “Terminological Entry” . “ <i>terminological entry languages</i> ” . “Language Section” . “name entry” . NOTES ⁽¹⁾⁽²⁾ The first "Administered Item" is the one which is being classified. The second "Administered Item" is the Classification Scheme itself.	N/A
Definition:	A reference to (a) class(es) of a scheme for the arrangement or division of	<i>Classification Scheme</i> : The descriptive information for an arrangement or division	The name of a particular arrangement or division of objects into groups based

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
	objects into groups based on characteristics that the objects have in common, e.g. origin, composition, structure, application, function etc.	of objects into groups based on characteristics which the objects have in common. <i>name</i> : A name by which an Administered Item (in this case the Classification Scheme) is known within a specific Context.	on characteristics which the objects have in common.
Obligation:	Optional	Conditional	Conditional
Condition:	N/A	If a Classification Scheme is used, its name is mandatory.	If a Classification Scheme is used, its name is mandatory.
Data type:	Character string	String	String
Comment	The definition does not specify whether the reference is by name or identifier.		

C.2.4.2 Classification scheme identifier

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Classification scheme	"Classification Scheme" . "classification scheme administration record" . "Administration Record" . "administered item identifier"	classification scheme identifier
Path from Administered Item:	N/A	"Administered Item" . " <i>administered item classification</i> " . "Classification Scheme Item" . " <i>classification scheme membership</i> " .	N/A
Definition:	A reference to (a) class(es) of a scheme for the arrangement or division of objects into groups based on characteristics that the objects have in common, e.g. origin, composition, structure, application, function etc.	<i>Classification Scheme</i> : The descriptive information for an arrangement or division of objects into groups based on characteristics which the objects have in common. <i>administered item identifier</i> : An identifier for an Administered Item (in this case the Classification Scheme) within a Registration Authority.	The identifier of a particular arrangement or division of objects into groups based on characteristics which the objects have in common.
Obligation:	Optional	Conditional	Optional
Condition	N/A	If a Classification Scheme is used, its administered item	N/A

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
		identifier is mandatory.	
Data type:	Character string	String	String
Comment	The definition does not specify whether the reference is by name or identifier.		

C.2.4.3 Classification scheme type name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Classification Scheme" . "classification scheme type name"	Classification scheme type name
Path from Administered Item:	N/A	"Administered Item" . " <i>administered item classification</i> " . "Classification Scheme Item" . " <i>classification scheme membership</i> " .	N/A
Definition:	N/A	The name of the type of classification scheme.	The name of the type of classification scheme.
Obligation:	N/A	Conditional	Optional
Condition	N/A	If Classification Scheme is present, <i>classification scheme type name</i> is mandatory.	N/A
Data type:	N/A	String	String

C.2.4.4 Classification scheme item type name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Classification Scheme Item" . "classification scheme item type name"	classification scheme item type name
Path from Administered Item:	N/A	"Administered Item" . " <i>administered item classification</i> " .	N/A
Definition:	N/A	The name of the type of the classification scheme item.	The name of the type of the classification scheme item.
Obligation:	N/A	Optional	Optional
Data type:	N/A	String	String

C.2.4.5 Classification scheme item value

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Keyword	"Classification Scheme Item" . "classification scheme item value"	classification scheme item value
Path from Administered Item:	N/A	"Administered Item" . "administered item classification" .	N/A
Definition:	One or more significant words used for retrieval of data elements.	An instance of a classification scheme item.	An instance of a classification scheme item.
Obligation:	Optional	Optional	Optional
Data type:	Character string	String	
Comment:	This attribute can be used for recording keywords (search keys) associated with the data element in question.	This edition of this part of ISO/IEC 11179 treats keywords as a type of classification scheme, with individual keywords being represented as classification scheme item values..	This edition of this part of ISO/IEC 11179 treats keywords as a type of classification scheme, with individual keywords being represented as classification scheme item values.

C.2.4.6 Related metadata reference

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Related data reference	"Administration Record" . "administered item identifier" OR "Reference Document" . "reference document identifier"	Related metadata reference
Path from Administered Item:	N/A	"Administered Item" . " <i>administered item relationship</i> " . Administered Item" . "administered item administration record" . OR "Administered Item" . "reference"	N/A
Definition:	A reference between the data element and any related data.	<i>administered item relationship</i> : the relationship among two or mote Administered Items. <i>administered item identifier</i> : An identifier for an Administered Item. <i>Reference Document</i> : a document that provides	A reference from one metadata item to another.

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
		pertinent details for consultation about a subject. <i>reference document identifier</i> : An identifier for the Reference Document.	
Obligation:	Optional	Optional	Optional
Data type:	Character string	String	String
Comment:	If this attribute occurs it shall be specified in pair with the attribute 'Type of relationship'	Within a single registry, Administered Items may be related through the Administered Item Relationship. If the reference needs to be external to the registry, a Reference Document may be used.	

C.2.4.7 Type of relationship

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Type of relationship	"Administered Item Relationship" . "administered item relationship type description" OR "Reference Document" . "reference document type description"	Type of relationship
Path from Administered Item:	N/A	"Administered Item" . " <i>administered item relationship</i> " OR "Administered Item" . "reference"	N/A
Definition:	An expression that characterizes the relationship between the data element and related data.	The description of the type of association with another data element concept that this data element concept modifies, is modified by, or is otherwise linked with.	The description of the type of relationship identified by the related metadata reference.
Obligation:	Conditional	Conditional OR Optional	Conditional
Condition:	This attribute is mandatory if the attribute 'related data reference' occurs.	"administered item relationship type description" is mandatory if the " <i>administered item</i>	This attribute is mandatory if the attribute 'related metadata reference' occurs.

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
		<i>relationship</i> " exists. "reference document type description" is optional if Reference Document is used..	
Data type:	Character string	String	String
Comment:	Examples of type of relationships are: 'qualifier of', 'qualified by', 'subject of', 'part of', 'physical condition', 'external reference', 'higher standard', 'data element concept'.	See C.2.4.6 Related metadata reference.	

C.2.5 Attributes specific to Data Element Concepts

C.2.5.1 Object class name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Designation (of Administered)" . "name" NOTE The Administered item referred to here is the Object Class.	Object class name
Path from Data Element Concept:	N/A	"Data Element Concept" . "data element concept object class" . "Object Class" . "object class administration record" . "Administration Record" . "Administered Item" . " <i>administered item context</i> " . "Terminological Entry" . " <i>terminological entry languages</i> " . "Language Section" . "name entry" .	N/A
Definition:	N/A	<i>data element concept object class</i> : the designation of an Object Class for a Data Element Concept. <i>name</i> : A name by which an Administered Item (in this case the Object Class) is known within a specific context.	The designation of an <i>object class</i> for a <i>data element concept</i> .
Obligation:	N/A	Optional	Optional
Data type:	N/A	String	String

C.2.5.2 Object class identifier

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Administration Record" . "administered item identifier" NOTE The Administered item referred to here is the Object Class.	Object class identifier
Path from Data Element Concept:	N/A	"Data Element Concept" . "data element concept object class" . "Object Class" . "object class administration record" .	N/A
Definition:	N/A	<i>administered item identifier</i> : An identifier for an Administered Item (in this case the Object Class) within a Registration Authority.	The identifier of an <i>object class</i> for a <i>data element concept</i> .
Obligation:	N/A	Optional	Optional
Data type:	N/A	String	String

C.2.5.3 Property name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Designation (of Administered)" . "name" NOTE The Administered item referred to here is the Property.	Property name
Path from Data Element Concept:	N/A	"Data Element Concept" . "data element concept property" . "Property" . "property administration record" . "Administration Record" . "Administered Item" . " <i>administered item context</i> " . "Terminological Entry" . " <i>terminological entry languages</i> " . "Language Section" . "name entry" .	N/A
Definition:	N/A	<i>data element concept property</i> : the designation of a Property for a Data Element Concept. <i>name</i> : A name by which an Administered Item (in this case the Property) is known	The designation of a <i>property</i> for a <i>data element concept</i> .

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
		within a specific context.	
Obligation:	N/A	Optional	Optional
Data type:	N/A	String	String

C.2.5.4 Property identifier

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Administration Record" . "administered item identifier" NOTE The Administered item referred to here is the Property.	Property identifier
Path from Data Element Concept:	N/A	"Data Element Concept" . "data element concept property" . "Property" . "property administration record" .	N/A
Definition:	N/A	<i>administered item identifier:</i> An identifier for an Administered Item (in this case the Property) within a Registration Authority.	The identifier of a <i>property</i> for a <i>data element concept</i> .
Obligation:	N/A	Optional	Optional
Data type:	N/A	String	String

C.2.6 Attributes specific to Data Elements

C.2.6.1 Representation category

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Representation category	Not supported.	Not supported.
Definition:	Type of symbol, character or other designation used to represent a data element.	N/A	N/A
Obligation:	Mandatory	N/A	N/A
Data type:	Character string	N/A	N/A
Comment:	The representation category shall be specified by the relevant standard. Examples of possible representation categories: - character representation		

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
	(ISO/IEC 646) - character/symbol representation (ISO registration no. 143) - bar coded representation (EIA-556) - graphical representation		

C.2.6.2 Representation class

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Form of representation	"Designation (of Administered)" . "name" NOTE The Administered item referred to here is the Representation Class.	Representation class
Path from Data Element:	N/A	"Data Element" . " <i>data element representation class</i> " . "Representation Class" . "Administration Record" . "Administered Item" . " <i>administered item context</i> " . "Terminological Entry" . " <i>terminological entry languages</i> " . "Language Section" . "name entry" .	N/A
Definition:	Name or description of the form of representation for the data element, e.g. 'quantitative value', 'code', 'text', 'icon'.	<i>Representation Class</i> : the classification of types of representations. <i>name</i> : A name by which an Administered Item (in this case the Representation Class) is known within a specific context.	The name of the class of representation of a data element.
Obligation:	Mandatory	Optional	Optional
Data type:	Character string	String	String
Comment:	1. See ISO/IEC 11179-2 for appropriate terms ('property words' or 'class words') to be used. 2. Example 1: For the data element named: 'country of origin code' this attribute contains: 'code'. 3. Example 2: For the data element: 'product description' this attribute	See 4.13.1.4 for a list of Representation Class terms.	See 4.13.1.4 for a list of Representation Class terms.

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
	contains: 'text'. 4. Example 3: For the data element: 'weight of consignment' this attribute contains: 'quantitative value'.		

C.2.6.3 Value domain name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not directly supported.	"Designation (of Administered)" . "name" NOTE The Administered item referred to here is the Value Domain.	value domain name
Path from Data Element:	N/A	"Data Element" . " <i>data element representation</i> " . "Value Domain" . "value domain administration record" . "Administration Record" . "Administered Item" . " <i>administered item context</i> " . "Terminological Entry" . " <i>terminological entry languages</i> " . "Language Section" . "name entry" .	N/A
Definition:	N/A	<i>Value domain</i> : A set of permissible values. It provides representation, but has no implication as to what data element concept the values may be associated with nor what the values mean. <i>name</i> : A name by which an Administered Item (in this case the Value Domain) is known within a specific context.	The name of the value domain that provides representation for the data element.
Obligation:	N/A	Mandatory	Optional
Data type:	N/A	String	String
Comment:	The closest equivalent is "permissible data element values" (see F.2.6.10), but this actually represents the values.		

C.2.6.4 Value domain identifier

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not directly supported.	"Administration Record" . "administered item identifier"	value domain identifier
Path from Data Element:	N/A	"Data Element" . " <i>data element representation</i> " . "Value Domain" . "value domain administration record" .	N/A
Definition:	N/A	<i>Value Domain</i> : A set of permissible values. It provides representation, but has no implication as to what Data Element Concept the values may be associated with nor what the values mean. <i>administered item identifier</i> : An identifier for an administered item (in this case the Value Domain) within a registration authority.	The identifier of the value domain that provides representation for the data element.
Obligation:	N/A	Mandatory	Optional
Data type:	N/A	String	String
Comment:	The closest equivalent is "permissible data element values" (see F.2.6.10), but this actually represents the values.		

C.2.6.5 Datatype name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Datatype of data element values	"Datatype" . "datatype name"	datatype name
Path from Data Element:	N/A	"Data Element" . " <i>data element representation</i> " . "Value Domain" . "value domain datatype" .	N/A

Definition:	A set of distinct values for representing the data element value.	<i>Datatype</i> : A set of distinct values characterized by properties of those values and by operations on those values. <i>datatype name</i> : A designation for the datatype.	<i>datatype name</i> : A designation for the datatype.
Obligation:	Mandatory	Mandatory	Conditional
Condition	N/A	N/A	Required if neither <i>value domain name</i> nor <i>value domain identifier</i> is specified.
Data type:	Character string	String	String
Comment:	Examples: Possible instances are: 'character', 'ordinal number', 'integer', 'real', 'scaled', 'bit', 'rational'. Note: The examples suggest the attribute is intended to be the name of the datatype, whereas the definition implies it is a set of values.	In the metamodel, the datatype is an attribute of the value domain, not directly of the data element.	

C.2.6.6 Datatype scheme reference

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Datatype" . "datatype scheme reference"	Datatype scheme reference
Path from Data Element:	N/A	"Data Element" . " <i>data element representation</i> " . "Value Domain" . "value domain datatype" .	N/A
Definition:	N/A	A reference identifying the source of the Datatype specification.	A reference identifying the source of the datatype specification.
Obligation:	N/A	Mandatory	Conditional
Condition	N/A	N/A	Required if <i>datatype name</i> is specified.
Data type:	N/A	String	String
Comment:		In the metamodel, the datatype is an attribute of the value domain, not directly of the data element.	

C.2.6.7 Maximum size

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Maximum size of data element values	“Value Domain” . “value domain maximum character quantity”	Maximum size
Path from Data Element:	N/A	“Data element” . “data element representation” .	N/A
Definition:	The maximum number of storage units (of the corresponding datatype) to represent the data element value.	The maximum number of characters to represent the data element value. NOTE Applicable only to character Datatypes.	The maximum number of storage units (of the corresponding datatype) to represent the data element value.
Obligation:	Mandatory	Optional	Optional
Data type:	Integer	Integer	Integer
Comment:	<p>1. Example 1:</p> <p>For data element: 'invoice number' the attribute 'datatype' has instance 'character'</p> <p>and the attribute 'maximum size of data element value' has value: '17'. The data element value of 'invoice number' shall have a maximum of 17 characters.</p> <p>2. The two attributes 'maximum and minimum (see 6.4.5) size of data element values' indicate whether data element values are 'fixed' (maximum and minimum size are equal) or 'variable' (maximum and minimum size vary).</p>	<p>This is not exactly equivalent, because it applies only to character datatypes.</p>	

C.2.6.8 Minimum size

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Minimum size of data element values.	Not supported.	Minimum size
Definition:	The minimum number of storage units (of the corresponding datatype) to represent the data element value.	N/A	The minimum number of storage units (of the corresponding datatype) to represent the data element value.
Obligation:	Mandatory	N/A	Optional
Data type:	Integer	N/A	Integer
Comment:	<p>1. Example 1:</p> <p>For data element: 'product description' the attribute 'datatype' has instance 'character' and the attribute 'minimum size of data element value' has instance: '10'.</p> <p>The data element value of 'product description' shall have a minimum of 10 characters.</p> <p>2. The two attributes 'maximum (see 6.4.4) and minimum size of data element values' indicate whether data element values are 'fixed' (maximum and minimum size are equal) or 'variable' (maximum and minimum size vary).</p>		

C.2.6.9 Layout of representation

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Layout of representation	Not supported.	Layout of representation
Definition:	The layout of characters in data element values expressed by a character string representation.	N/A	The layout of characters in data element values expressed by a character string representation.
Obligation:	Conditional	N/A	Optional
Condition:	If the data element is of the class 'quantitative data' this attribute is mandatory. If the		

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
	attribute 'form of representation' is 'code' the use of this attribute is recommended if the code representation has to have a specific structure or layout.		
Data type:	Character string		String
Comment:	<p>1. For quantitative data it is necessary to distinguish between integers, decimal mark and floating point notations.</p> <p>Example:</p> <p>Integers may be indicated with 'n', for decimal mark the number of characters before and after the decimal mark are specified as: n(5).n(3), for floating point notations the layout convention for a value with exponents shall comply with ISO 6093: n(3).n(3)E2, where 'E2' stands for max. 2 digits for the power of 10.</p> <p>2. For code representations having a specific structure or layout the type of character for each position in the code structure is important for validation purposes.</p> <p>Example:</p> <p>The data element 'flight number' has an international code representation structure consisting of two alphabetic characters of the airline company followed by a three-digit number identifying the flight (from starting-point to destination).</p> <p>The contents of the attribute: 'layout of representation' is: 'AA999'.</p>		

C.2.6.10 Permissible data element values

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Permissible data element values	See Value Domain for equivalent capability.	See Value Domain for equivalent capability.
Definition:	The set of representations of permissible instances of the data element, according to the representation form, layout, datatype and maximum and minimum size specified in the corresponding attributes. The set can be specified by name, by reference to a source, by enumeration of the representation of the instances or by rules for generating the instances.	N/A	N/A
Obligation:	Mandatory	N/A	N/A
Data type:	Character string	N/A	N/A
Comment:	When the permissible data element values are an enumeration of coded representations each data element value and instance shall be presented as a pair.		

C.2.7 Attributes specific to Conceptual Domains

C.2.7.1 Dimensionality

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Conceptual Domain" . "dimensionality"	dimensionality
Definition:	N/A	The dimensionality for a concept.	The dimensionality for a concept.
Obligation:	N/A	Optional	Optional
Data type:	N/A	String	String
Comment:		For example, length, mass, velocity, currency.	For example, length, mass, velocity, currency.

C.2.8 Attributes specific to Value Domains

C.2.8.1 Datatype name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	See "Datatype of data element values" (F.2.6.5)	"Value Domain" . "value domain datatype" . "Datatype" . "datatype name"	datatype name
Definition:	N/A	<i>Datatype</i> : A set of distinct values characterized by properties of those values and by operations on those values. <i>datatype name</i> : A designation for the datatype.	<i>datatype name</i> : A designation for the datatype.
Obligation:		Mandatory	Mandatory
Data type:		String	String

C.2.8.2 Datatype scheme reference

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Value Domain" . "value domain datatype" . "Datatype" . "datatype scheme reference"	Datatype scheme reference
Definition:		A reference identifying the source of the datatype specification.	A reference identifying the source of the datatype specification.
Obligation:		Mandatory	Optional
Data type:		String	String

C.2.8.3 Unit of measure name

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Value Domain" . "value domain unit of measure" . "Unit of Measure" . "unit of measure name"	unit of measure name
Definition:		The name of a unit of measure.	The name of a unit of measure.
Obligation:		Optional	Optional
Data type:		String	String

C.2.9 Attributes specific to Permissible Values

C.2.9.1 Value

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	See "permissible data element values" (F.2.6.10)	"Permissible Value" . "permitted value" . "Value" . "value item"	value
Definition:	N/A	A representation of a value meaning in a specific value domain. The actual value.	A representation of a value meaning in a specific value domain. The actual value.
Obligation:	N/A	Mandatory	Mandatory
Data type:	N/A	String	String
Comment:			

C.2.9.2 Permissible Value Begin Date

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Permissible Value" . "permissible value begin date"	permissible value begin date
Definition:	N/A	The date this value became/becomes permissible in the value domain.	The date this value became/becomes permissible in the value domain.
Obligation:	N/A	Optional	Optional
Data type:	N/A	Date	Date

C.2.9.3 Permissible Value End Date

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Permissible Value" . "permissible value end date"	permissible value end date
Definition:	N/A	The date this value became/becomes no longer permissible in the value domain.	The date this value became/becomes no longer permissible in the value domain.
Obligation:	N/A	Optional	Optional
Data type:	N/A	Date	Date

C.2.10 Attributes specific to Value Meanings

C.2.10.1 Value meaning description

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Value Meaning" . "value meaning description"	value meaning description
Definition:	N/A	A description of a value meaning.	A description of a value meaning.
Obligation:	N/A	Mandatory	Mandatory
Data type:	N/A	String	String

C.2.10.2 Value meaning identifier

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Value Meaning" . "value meaning identifier"	value meaning identifier
Definition:	N/A	The unique identifier for a value meaning.	The unique identifier for a value meaning.
Obligation:	N/A	Mandatory	Optional
Data type:	N/A	String	String

C.2.10.3 Value meaning begin date

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Value Meaning" . "value meaning begin date"	value meaning begin date
Definition:	N/A	The effective date of this value meaning in the conceptual domain.	The effective date of this value meaning in the conceptual domain.
Obligation:	N/A	Optional	Optional
Data type:	N/A	Date	Date

C.2.10.4 Value meaning end date

	<u>1994 clause 6</u>	<u>2002 clause 4</u>	<u>2002 clause 5</u>
Attribute name:	Not supported.	"Value Meaning" . "value meaning end date"	value meaning end date
Definition:	N/A	The date this value meaning became/becomes invalid.	The date this value meaning became/becomes invalid.
Obligation:	N/A	Optional	Optional

Data type:	N/A	Date	Date
------------	-----	------	------

Bibliography

- [1] Flavin, Matt. *Fundamental Concepts of Information Modeling*. Yourdon Press; 1981.
- [2] Fowler, Martin. *UML Distilled*. Addison Wesley Publishing Company: 1997.
- [3] ISO/IEC 10027:1990, *Information technology – Information Resource Dictionary System (IRDS) Framework*
- [4] ISO/IEC 11404:1995, *Information technology – Language-Independent Datatypes*
- [5] ISO/IEC TR 15452:2000, *Information technology – Specification of data value domains*
- [6] National Institute of Standards and Technology Special Publication 500-208, *Manual for Data Administration*, edited by Judith J. Newton and Daniel C. Wahl, National Institute of Standards and Technology, 1993.