TECHNICAL SPECIFICATION

ISO/TS 21526

First edition
2019-10

Health informatics — Metadata repository requirements (MetaRep)

Informatique de santé — Exigences relatives aux référentiels de métadonnées (MetaRep)

Reference number
ISO/TS 21526:2019(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 215, *Health informatics*.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

This document is intended to be an extension to and a clarification of the ISO/IEC 11179 series.

Healthcare has a fundamental requirement for describing the meaning, provenance and governance of data, and for setting standards for how that data is stored and communicated. Unsurprisingly, it uses metadata registries and repositories extensively for a wide range of purposes supporting care delivery, reporting and research. However, these registries are only partially interoperable, with consequences to cost of ownership and utility that lead to under-use, particularly where simple, read-only repositories are required. There is also considerable unmet need resulting from the community's focus on message-based interoperability at the expense of the description of the meaning of the data in the systems that are the source of that data, and how that data maps to the meaning of the large quantities of data communicated between providers and national bodies in tables or simple XML.

Data in healthcare systems should persist in content and meaning across organizations and time for a wide variety of uses including patient care, patient safety, service management, service improvement, public health and healthcare research. The sharing and adoption of record or message designs offer immediate and tangible benefits to these ends, entailing organizations to adopt common standards for the exchange of the specifications of records and messages in terms of the representation and definition of individual elements of data; the inter-relationships of those elements in data models and where sets of data that accord to those models can be found together with any contextual information about those data sets that is required for their understanding and appropriate reuse.

Settings where metadata collections are assembled include individual clinics and hospitals, organizations managing a portfolio of clinical studies, organizations providing cloud applications in support of healthcare and regional and national bodies who commission standard reports and datasets in pursuit of policy objectives. The intent is to support an ecosystem of interoperable registries and repositories which facilitate both the development and implementation of content standards – and thus interoperable content – and the publication of interoperable metadata about the kinds of data available in both care and research.

This document includes a review of existing components of ISO/IEC 11179-3, i.e. Metadata Registry (MDRMetamodel), and ISO/IEC 19763-12 Metamodel for Information Models (MFIInformationModel) to specify where variations from or additions to the requirements of these standards are required for specific healthcare use cases. Registries conforming to this document are also likely to reference of ISO/IEC 11179-7[1] Metamodel for dataset registration (MDRDatasets) and ISO/IEC TS 19763-13 Metamodel for forms registration (MFIForms), however it is less clear that simplifications or extensions to either are necessary in the healthcare or healthcare research setting and thus the original documents should be used as is.

MDRMetamodel provides a comprehensive model for an international metadata registry addressing several large communities with overlapping concerns, and the conformance statements in the 2013 edition are framed in this context. Equally MFIInformationModel is designed to represent models represented in many ways from purely conceptual entity relationship diagrams through to a concrete relational database instance. A metadata registry/repository aimed at a less diverse community such as healthcare or directed at the needs of a smaller organisation might not require the complete implementation of ISO/IEC 11179-3 and ISO/IEC 19763-12, so it is important that any restriction or simplification is shared to preserve registry interoperability.

[1] Under preparation. Stage at the time of publication: ISO/IEC/PRF 11179-7:2019.

Health informatics — Metadata repository requirements (MetaRep)

1 Scope

This document describes requirements for collections of metadata about data elements and their containing models and datasets in a healthcare environment. The collection can serve as a repository or as dictionary describing a set of items in use in a particular domain, organisation or product for reference, or it can additionally serve as a registry, supporting the development and communication of standard items to an audience with shared goals.

This document specifies standard refinements that account for the detailed governance and administration requirements that are particular to healthcare data. It focuses on the description of data that is persisted in healthcare systems rather than the specification of messages between these systems. It describes necessary extensions to the ISO/IEC 11179 series and to other International Standards on metadata originating from ISO/IEC JTC 1/SC 32 to address the wider variety of value domain types found in healthcare. Where appropriate, it also suggest restrictions/simplifications to the ISO/IEC 11179 series that promote wider adoption without compromising interoperability between metadata registries and opportunities for the development of tooling that consumes metadata for the generation or the parameterization of code.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

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

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1
acceptability rating
scale of acceptability

3.2
administered item
registered item for which *administrative information* (3.3) is recorded

3.3
administrative information
<metadata registry> information about the administration of an item in a *metadata registry* (3.33)

EXAMPLE Creation date, last change date, origin, change description, explanatory comment.

3.4
administration record
collection of *administrative information* (3.3) for an *administered item* (3.2)

3.5
administrative status
designation of the status in the administrative process of a *registration authority* (3.42) for handling *registration* (3.41) requests

Note 1 to entry: The values and associated meanings of "administrative status" are determined by each registration authority.

3.6
attribute
characteristic (3.12) of an object or entity

3.7
assertion
sentence or proposition in logic which is asserted (or assumed) to be true

3.8
attribute instance
specific instance of an *attribute* (3.6)

Note 1 to entry: Adapted from ISO 2382-17:1993 to distinguish an instance of an attribute from its value.

[SOURCE: ISO/IEC 11179-3:2013, 3.2.7]

3.9
binding
mapping from one framework or specification to another, enabling *data* (3.20) and/or commands to be passed between them

3.10
boolean
mathematical *datatype* (3.23) associated with two-valued logic

[SOURCE: ISO/IEC 11179-3:2013, 3.2.12]

3.11
cardinality
number of elements in a set

3.12
characteristic
abstraction of a *property* (3.39) of an object or of a set of objects

Note 1 to entry: Characteristics are used for describing concepts.

[SOURCE: ISO/IEC 11179-3:2013, 3.2.14]

3.13
class
description of a set of objects that share the same *attributes* (3.6), operations, methods, *relationships* (3.44), and semantics

3.14
classification scheme
descriptive information for an arrangement or division of objects into groups based on *characteristics* (3.12), which the objects have in common

3.15
concept
unit of knowledge created by a unique combination of *characteristics* (3.12)

[SOURCE: ISO/IEC 11179-3:2013, 3.2.18]

3.16
concept system
set of *concepts* (3.15) structured according to the relations among them

[SOURCE: ISO/IEC 11179-3:2013, 3.2.19]

3.17
conceptual domain
CD
concept (3.15) that expresses its description or valid instance meanings

Note 1 to entry: The value meanings may either be enumerated or expressed via a description.

[SOURCE: ISO/IEC 11179-3:2013, 3.2.21]

3.18
contact
instance of a role of an individual or *organization* (3.37) (or organization part or organization person) to or from whom an information item(s), a material object(s) and/or person(s) can be sent in a specified *context* (3.19)

3.19
context
circumstance, purpose, and perspective under which an object is defined or used

3.20
data
re-interpretable representation of facts, *concepts* (3.15), or instructions in a formalized manner suitable for communication, interpretation, or processing

Note 1 to entry: Data can be processed by human or automatic means.

[SOURCE: ISO/IEC 11179-3:2013, 3.2.27]

3.21
data element
DE
unit of *data* (3.20) for which the *definition* (3.28), identification, representation and *permissible values* (3.38) are specified by means of a set of *attributes* (3.6)

3.22
data element concept
DEC
concept (3.15) that can be represented in the form of a *data element* (3.21), described independently of any particular representation

3.23
datatype
set of distinct *values* (3.46), characterized by properties of those values and by operations on those values

[SOURCE: ISO/IEC 11179-3:2013, 3.1.9]

3.24
dimensionality
expression of measurement without units

Note 1 to entry: A quantity is a value with an associated unit of measure. 32º Fahrenheit, 0º Celsius, $100 USD, and 10 reams (of paper) are quantities. Equivalence between two units of measure is determined by the existence of a quantity preserving one-to-one correspondence between values measured in one unit of measure and values measured in the other unit of measure, independent of context, and where characterizing operations are the same. Equivalent units of measure in this sense have the same dimensionality. The equivalence defined here forms an equivalence relation on the set of all units of measure. Each equivalence class corresponds to a dimensionality. The units of measure "temperature" in degrees Fahrenheit" and "temperature in degrees Celsius" have the same dimensionality, because for each value measured in degrees Fahrenheit there is a value measured in degrees Celsius with the same quantity, and vice-versa. The same operations may be performed on quantities in each unit of measure. Quantity preserving one-to-one correspondences are the well-known equations Cº = (5/9)*(Fº – 32) and Fº = (9/5)*(Cº) + 32.

3.25
enumerated conceptual domain
conceptual domain that is specified by a list of all its *value meanings* (3.48)

3.26
enumerated value domain
value domain (3.47) that is specified by a list of all its *permissible values* (3.38)

3.27
data element value
value out of a set of *permissible values* (3.38) pertaining to a *data element* (3.21)

3.28
definition
representation of a *concept* (3.15) by a descriptive statement which serves to differentiate it from related concepts

[SOURCE: ISO/IEC 11179-3:2013, 3.2.29]

3.29
identifier
<metadata registry> sequence of characters, capable of uniquely identifying that with which it is associated, within a specified *context* (3.19)

Note 1 to entry: A name should not be used as an identifier because it is not linguistically neutral.

3.30
metadata
data that defines and describes other *data* (3.20)

[SOURCE: ISO/IEC 11179-3:2013, 3.2.74]

3.31
metadata item
instance of a *metadata object* (3.32)

3.32
metadata object
object type defined by a metamodel

3.33
metadata registry
MDR
information system for registering *metadata* (3.30)

[SOURCE: ISO/IEC 11179-3:2013, 3.2.78]

3.34
name
primary means of identification of objects and *concepts* (3.15) for humans

3.35
object class
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

3.36
ontology
a conceptualisation of a domain

3.37
organization
<management> unique framework of authority within which a person or persons act, or are designated to act, towards some purpose

[SOURCE: ISO/TS 21089:2018, 3.97]

3.38
permissible value
expression of a *value meaning* (3.48) allowed in a specific *value domain* (3.47)

3.39
property
characteristic (3.12) common to all members of an *object class* (3.35)

3.40
registrar
representative of a *registration authority* (3.42)

3.41
registration
relationship (3.44) between an *administered item* (3.2) and the *registration authority* (3.42)

3.42
registration authority
RA
organization (3.37) responsible for maintaining a register

3.43
registration status
designation of the status in the *registration* (3.41) life-cycle of an *administered item* (3.2)

3.44
relationship
connection among model elements

3.45
unit of measure
<data> actual units in which the associated *values* (3.46) are measured

Note 1 to entry: The dimensionality of the associated conceptual domain shall be appropriate for the specified unit of measure.

3.46
value
data (3.20) *value* (3.46)

3.47
value domain
value set
set of *permissible values* (3.38)

Note 1 to entry: 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 2 to entry: The permissible values may either be enumerated or expressed via a description.

[SOURCE: ISO/IEC 11179-3:2013, 3.2.140 — modified, "value set" was added as a preferred term.]

Note 3 to entry: 'Value set' is Health Level 7 (HL7) terminology for value domain, as defined in Reference [13].

3.48
value meaning
meaning or semantic content of a *value* (3.46)

4 Framework for the management of metadata registry content

4.1 Overall approach

The ISO/IEC JTC 1/SC 32 International Standards portfolio on metadata provides a modular approach to the registration of metadata about a wide variety of models and the management of that registry. MDRMetamodel consists of two distinct components: common facilities defining metadata about content – metadata objects – with attributes for identification, naming, definition, classification, conceptualisation, administration and registration; and those defining content models for some of the metadata objects to be registered – concept systems, value lists and data elements. These common facilities are reused in MFIInformationModel for the registration of information models, MDRDatasets for datasets and MFIForms for form designs. A conceptual representation of the relationship between the components and standards is presented in Figure 1 where dependencies between standards are indicated by vertical layering. MDRMetamodel, MFIInformationModel, MDRDatasets and MFIForms cover a wide range of usecases, not all of which are necessary for application to healthcare.

The approach taken in this document is to distil minimal requirements from the portfolio that shall be met by compliant registries, present these as a consistent, unified metamodel, and then extend these minimal requirements for the specific requirements in healthcare. While the UML class diagram representation of the models is retained, a minimal subset of the UML is used in the presentation of the models, and the modelling style has been altered to facilitate translation into relational or document (XML/JSON) implementation. Nevertheless all models contained within this document are iso-semantic with those in the source standards, and registries compliant with this this document will additionally be able to claim basic compliance with MDRMetamodel, MFIInformationModel, MDRDatasets and MFIForms.

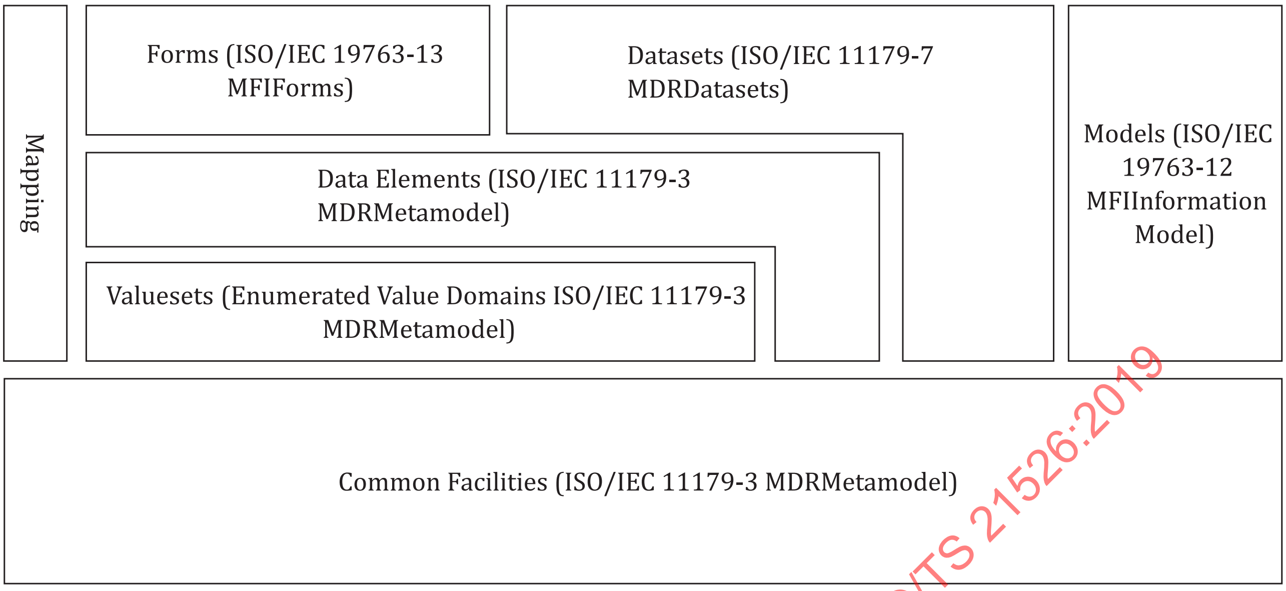
Mapping
Forms (ISO/IEC 19763-13 MFIForms)
Datasets (ISO/IEC 11179-7 MDRDatasets)
Models (ISO/IEC 19763-12 MFIInformation Model)
Data Elements (ISO/IEC 11179-3 MDRMetamodel)
Valuesets (Enumerated Value Domains ISO/IEC 11179-3 MDRMetamodel)
Common Facilities (ISO/IEC 11179-3 MDRMetamodel)

Figure 1 — Relationships between ISO/IEC JTC 1/SC 32 International Standards on metadata

The models in MDRMetamodel, MFIInformationModel, MDRDatasets and MFIForms are conceptual – there is no requirement that the specified model structures should be faithfully implemented in the actual schema of the registry/repository itself, nor any requirement to adopt either a relational, hierarchical or graph approach in their implementation, even though the literal interpretation of MDRMetamodel is most easily realised in a graph database. Relational implementations might implement separate schemas for administrative and semantic functions, provided the relevant administration record is linked to a class that is specified as a kind of administered item.

It is also worth noting that the basic data types described in MDRMetamodel, MFIInformationModel, MDRDatasets and MFIForms are those required for the implementation of the standard itself, not of the content of the registry/repository – MDRMetamodel has facilities for declaring specific datatypes of interest to a particular registry/repository's users, including those that are specific to healthcare information systems.

From this point on, registry and repository are used interchangeably unless specific reference is made. Implementers of simple repositories can choose not to implement those parts of the administrative elements of the common facilities which includes support the content creation lifecycle.

The Metamodel presented here is split into six sections: basic types; common facilities; data description; model description; schema registration; mapping, each of which are described below in textual form with illustrative, non-normative diagrams.

For a detailed discussion of how MDRMetamodel, MDRInformationModel have been restricted and extended in the derivation of this document, see Annex A.

4.2 Basic Types

4.2.1 General

Basic types contain foundational datatypes used in the metamodels described in this document.

4.2.2 BLOB (or binary large object)

A large block of binary data, typically an image or video file, that might have to be handled in a special way.

4.2.3 Boolean

A mathematical datatype associated with two-valued logic (see ISO/IEC 11404:2007, 8.1.1).

NOTE The notation and semantics for boolean are as described in ISO/IEC 11404.

4.2.4 Date

A datatype whose values are points in time to the resolution: year, month, and day (see ISO/IEC 11404:2007, 8.1.6).

4.2.5 Datetime

A datatype whose values are points in time to the resolution: year, month, day, hour, minute, second, and optionally fractions of seconds (see ISO/IEC 11404:2007, 8.1.6).

4.2.6 Integer

A mathematical datatype comprising the exact integral values (see ISO/IEC 11404:2007, 8.1.7).

NOTE Both the notation and semantics of the Integer datatype is as specified in ISO/IEC 11404:2007, 8.1.7.

4.2.7 Phone_Number

A phone number uniquely identifies a telephone line within a telephone network. The data structure of the Phone_Number data element shall conform to ITU-T E 164.

4.2.8 Postal_Address

A postal address enables the unambiguous determination of an actual or potential delivery point, usually combined with the specification of an addressee and/or a mailee.

4.2.9 String

A datatype comprising of a serial sequence of characters, bytes, integers, etc. See ISO 12639:2004, 4.1.11.

4.3 Common Facilities

4.3.1 General

Common facilities (see Figure 2) provides the model for an Administered Item, a registered item for which administrative information is recorded. Administered items are significant metadata entries that are individually identified, named, defined, classified, described by reference documents and have status with respect to registration and administration managed within the registry. Requirements for common facilities are derived from MFIMetamodel, but apply to all kinds of metadata item described in this document.

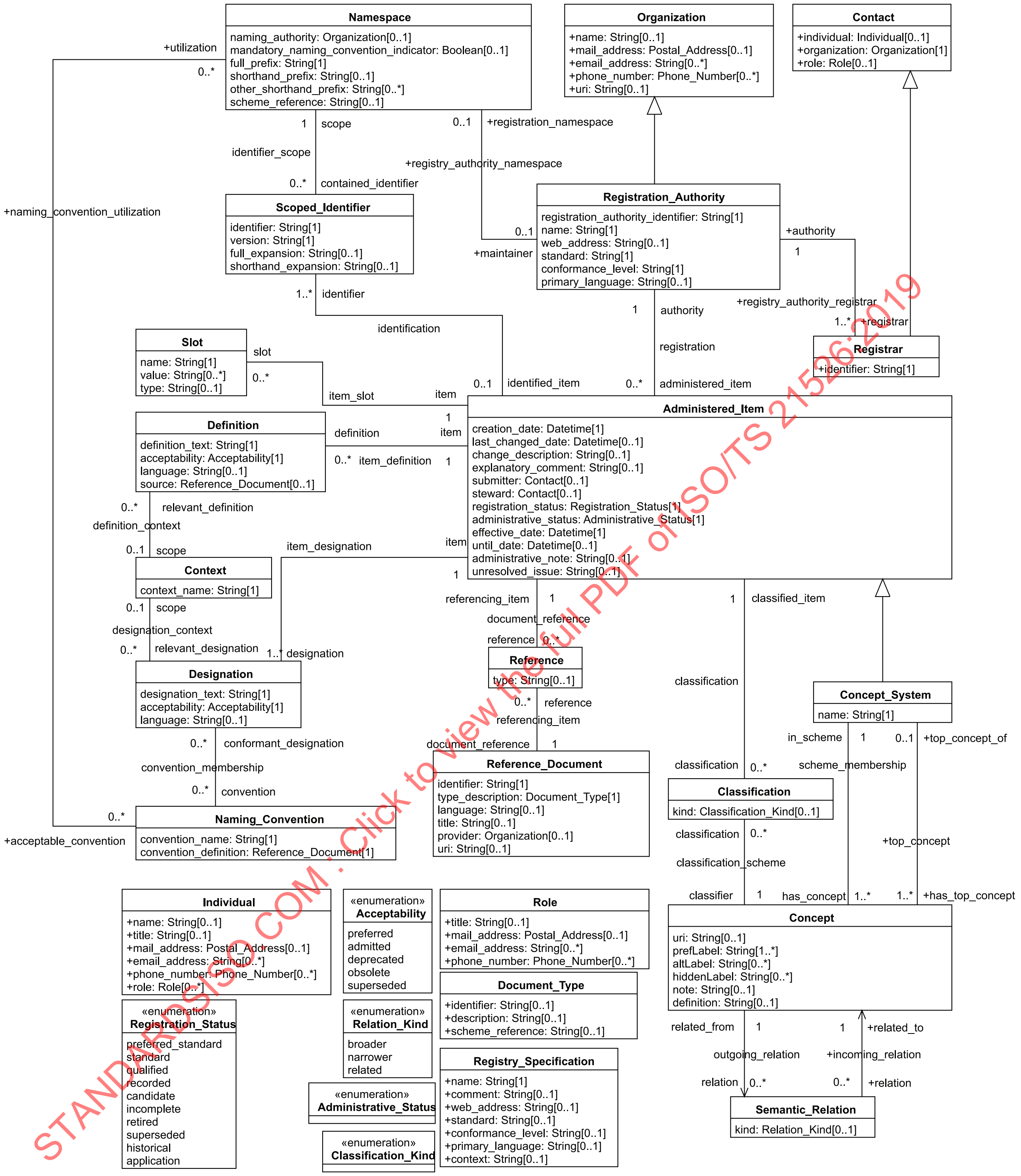

Figure 2 — Common facilities

4.3.2 Acceptability

Acceptability models a scale of acceptability ratings comprised of: preferred, admitted, deprecated, obsolete and superseded. This enumeration is used as a datatype for the attributes Designation. acceptability and Definition.acceptability.

Enumeration	Meaning
preferred	The designation or definition is promoted as the standard way of referring to the administered item in the referenced context
admitted	The designation or definition may be used as an alternative in relation to the administered item in the referenced context
deprecated	The designation or definition was once preferred or admitted in the referenced context, but is no longer so
obsolete	The designation or definition should not be used in the referenced context
superseded	The designation or definition has been replaced within the referenced context

4.3.3 Administered_Item

Administered_Item is a class, each instance of which models an administered item, an item in a metadata registry for which administrative metadata is required. An Administered_Item in metarep combines a subset of attributes, classes and relationships from Identified_Item, Registered_Item, Designatable_ Item, Administered_Item and Classifiable_Item classes in MDRMetamodel.

Attribute	Datatype	Multiplicity	Description
creation_date	Datetime	1	Date the Administered_Item was created
last_changed_date	Datetime	0..1	Date the Administered_Item was last changed
change_description	String	0..1	Description of what was changed since the prior version of the Administered_Item
explanatory_comment	String	0..1	Descriptive comments about the Administered_Item
submitter	Contact	0..1	The Contact who submitted this content to the metadata registry
steward	Contact	0..1	The Contact responsible for the submission of the Administered_Item to the registry
registration_status	Registration_Status	1	Designation of the status in the registration lifecycle of an Administered_Item
administrative_status	Administrative_Status	1	Designation of the status in the administration lifecycle of an Administered_Item
effective_date	Datetime	1	Date and time an Administered_Item became available to registry users
until_date	Datetime	0..1	Date and time the registration of an Administered_Item is no longer effective
administrative_note	String	0..1	General notes about the registration
unresolved_issue	String	0..1	Any problem that remains unresolved regarding proper documentation of an Administered_Item

Reference	Class	Multiplicity	Description	Inverse
reference	Reference	0..*	References applicable to an Administered_Item	referencing_item
classification	Classification	0..*	Classifications made of the Administered_Item	classified_item
slot	Slot	0..*	Slots which extend an Administered_Item	item

identifier	Scoped_Identifier	1..*	Scoped_Identifiers for the Administered_Item within a specified Namespace	identified_item
definition	Definition	0..*	Definitions appropriate to an Administered Item	item
designation	Designation	1..*	Designations for an Administered_Item	item
authority	Registration_Authority	1	The Registration_ Authority responsible for the registration of an Administered_Item	administered_item
map_source	Map	0..*	Map instances for which an administered item is a source	source_item
map_target	Map	0..*	Map instances for which an administered item is a target	target_item

4.3.4 Administrative_Status

The designation of the stage of development and review of an Administered Item as it transitions between registration statuses. Registries shall specify a curation workflow for its content together with appropriate administrative statuses, but no particular workflow or statuses are required by this document.

4.3.5 Classification

The Classification association class is used to record the classification of an Administered_Item into a group designated by a Concept in a Concept_System. The exact semantics of the Classification association are not specified by this document but will depend upon way in which the classification scheme is used. For example, the Classification association might signify either an "is-a" or an "instance-of" relationship.

Attribute	**Datatype**	**Multiplicity**	**Description**	
kind	Classification_Kind	0..1	Classifier for an association between an administered item and a concept	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
classifier	Concept	1	The Concept referenced by a Classification	classification
classified_item	Administered_Item	1	The Administered_Item that is classified	classification

4.3.6 Classification_Kind

A type enumerating permitted relationship kinds between a classification concept and an administered item. Kinds might encompass semantic relationships - set membership, instantiation - or ones designed to describe their visualisation - tree, tag cloud, timeline.

4.3.7 Concept

Concept is a class each instance of which models a concept, a unit of knowledge created by a unique combination of characteristics. A concept is independent of representation. MDRMetamodel has been simplified for this document and is modelled according to the Simple Knowledge Organisation System (SKOS).

Attribute	**Datatype**	**Multiplicity**	**Description**	
uri	String	0..1	Uri for the concept for imported classification schemes	
prefLabel	String	1..*	Preferred lexical label for a concept	
altLabel	String	0..*	Other lexical label for a concept	
hiddenLabel	String	0..*	Lexical label for a concept that should be hidden when generating visual displays of the concept but should still be accessible to free text search operations.	
note	String	0..1	General note for any purpose	
definition	String	0..1	Statement or formal explanation of the meaning of a concept	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
relation	Semantic_Relation	0..*	Semantic_Relations a for which a Concept is a source	related_to
relation	Semantic_Relation	0..*	Semantic_Relations for which a Concept is the target	related_from
classification	Classification	0..*	Classifications of Administered_Items assigned to a Concept	classifier
in_scheme	Concept_System	1	The Concept_System to which a Concept belongs	has_concept
top_concept_of	Concept_System	0..1	The Concept_System for which a Concept is optionally a top concept for	has_top_concept

4.3.8 Concept_System

Concept_System is a class each instance of which models a concept system, a set of concepts structured according to the relations among them. MDRMetamodel Concept_System has been simplified for this document: terminologies are maintained separately in health care allowing this document, i.e. ISO/TS 21526, to be more specifically focussed on the navigation and organisation of metadata registry content.

Superclass: Administered_Item

Attribute	**Datatype**	**Multiplicity**	**Description**	
name	String	1	Name of the Concept_System	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
has_concept	Concept	1..*	Concepts contained in a Concept_System	in_scheme
has_top_concept	Concept	1..*	Top level Concepts for a concept system	top_concept_of

4.3.9 Contact

Contact is a class each instance of which models a contact, which specifies a role and/or an individual within an organization or an organization part to whom information item(s), material object(s) and/or person(s) can be sent to or from. Registrar is a subclass of Contact.

Attribute	**Datatype**	**Multiplicity**	**Description**
individual	Individual	0..1	Individual that is a contact
organization	Organization	1	Organization for which the Contact acts as a representative
role	Role	0..1	Specified responsibilities of the Contact

4.3.10 Context

Context is a class each instance of which models a context, the setting in which Designations are used to designate and Definitions are used to define a set of Administered_Items. Each Administered_Item might be designated and/or defined within one or more Contexts. A Context defines the setting within which the subject data has meaning. A Context might 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 registration authority responsible for the registry.

Attribute	**Datatype**	**Multiplicity**	**Description**	
context_name	String	1	Name of the Context	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
relevant_designation	Designation	0..*	Designations that are relevant within a Context	scope
relevant_definition	Definition	0..*	Definitions that are relevant within a Context	scope

4.3.11 Definition

The Definition class records the binding of a definition_text with its language to an Administered_Item according to an acceptability rating. Each Definition is situated with respect to a Context.

Attribute	**Datatype**	**Multiplicity**	**Description**	
definition_text	String	1	Text of the Definition	
acceptability	Acceptability	1	Acceptability rating of a Definition in a specified Context	
language	String	0..1	Language and variant in which the Definition is expressed	
source	Reference_Document	0..1		
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
Item	Administered_Item	1	The Administered_Item to which a Definition applies	definition
scope	Context	0..1	The Context within which a definition_text in a particular language is accepted according to the acceptability rating	relevant_definition

4.3.12 Designation

The Designation class records the binding of a designation_text with its language to an Administered_Item according to an acceptability rating. Each Designation is situated with respect to a Context, a Naming_Convention, and a Namespace.

Attribute	**Datatype**	**Multiplicity**	**Description**	
designation_text	String	1	Text of the Designation	
acceptability	Acceptability	1	Acceptability rating of a Designation in a specified Context	
language	String	0..1	Language and variant in which the Designation is expressed EXAMPLE en-GB.	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
item	Administered_Item	1	The Administered_Item for which a Designation applies	designation
convention	Naming_Convention	0..*	The Naming Convention to which a Designation conforms	conformant_designation

scope	Context	0..1	The Context within which a designation_text in a particular language is accepted according to an acceptability rating	relevant_designation

4.3.13 Document_Type

The composite datatype Document_Type is used to specify the document type of a Reference_Document. The document type can be specified using an identifier or a description.

Attribute	Datatype	Multiplicity	Description
identifier	String	0..1	Identifies the type of document
description	String	0..1	Describes the type of document
scheme_reference	String	0..1	Identification scheme from which the identifier and/or description are drawn

4.3.14 Individual

Individual is a class each instance of which models an individual, a single human being.

Attribute	Datatype	Multiplicity	Description
name	String	0..1	Name of the individual
title	String	0..1	Name of the position held by the individual
mail_address	Postal_Address	0..1	Postal address of the individual
email_address	String	0..*	Email address of the individual
phone_number	Phone_Number	0..*	Phone number of the individual
role	Role	0..*	Specified responsibilities of the individual

4.3.15 Namespace

Namespace is a class each instance of which represents a namespace. Namespace is a scoping construct used to group sets of Scoped_Identifiers used in a metadata registry. Distinct Namespaces permit independent development of metadata collections and/or ontologies. They permit enforcement of uniqueness constraints on identifiers within a specific Namespace without central coordination.

Attribute	Datatype	Multiplicity	Description
naming_authority	Organization	0..1	Organisation that has the authority for naming in the Namespace
mandatory_naming_convention_indicator	Boolean	0..1	Indicator specifying whether all Designations in this Namespace shall conform to one of the acceptable Naming_Conventions.
full_prefix	String	1	The designation of the namespace in full
shorthand_prefix	String	0..1	Prefix conventionally used as shorthand for a namespace, for greater readability, in text for human consumption.
other_shorthand_prefix	String	0..*	Other shorthand prefixes used for this Namespace.
scheme_reference	String	0..1	Reference identifying the type of the Namespace specification. For XML Namespaces, specify: http://www.w3.org/TR/1999/REC-xml-names-19990114/ For UML Namespaces, specify: http://www.omg.org/spec/UML/2.4.1/Core/Namespace

Reference	Class	Multiplicity	Description	Inverse
contained_identifier	Scoped_Identifier	0..*	Scoped_Identifiers within the Namespace	scope
maintainer	Registration_Authority	0..1	Registration_Authority for the Namespace applies	registration_namespace
acceptable_convention	Naming_Convention	0..*	Naming_Convention permitted for use within a Namespace	utilization

4.3.16 Naming_Convention

The Naming_Convention class provides the specification by which the text of a Designation is developed.

Attribute	Datatype	Multiplicity	Description	
convention_name	String	1	Name of the Naming_Convention	
convention_definition	Reference_ Document	1	Reference document which describes authority, semantic, syntactic and lexical rules which govern the generation of conformant designations	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
conformant_designation	Designation	0..*	Conformant Designations for a Naming_Convention	convention
utilization	Namespace	0..*	Namespaces to which a Naming_Convention applies	acceptable_convention

4.3.17 Organization

Organization is a class each instance of which models an organization, a unique framework of authority within which individuals act, or are designated to act, towards some purpose.

Attribute	Datatype	Multiplicity	Description
name	String	0..1	Name of the organisation
mail_address	Postal_Address	0..1	Postal address of the organisation
email_address	String	0..*	Email address of the organisation
phone_number	Phone_Number	0..*	Phone number of the organisation
uri	String	0..1	Uri of an Organisation where they might be contacted via the web

4.3.18 Reference

An association between a Reference_Document and an Administered_Item

Attribute	Datatype	Multiplicity	Description	
type	String	0..1	Specification of the type of Reference	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
document_reference	Reference_Document	1	The Reference_Document specified by a Reference	reference
referencing_item	Administered_Item	1	Administered_Items to which a Reference is appropriate	reference

4.3.19 Reference_Document

A Reference_Document is a document that provides pertinent details for consultation about a subject.

Attribute	Datatype	Multiplicity	Description	
identifier	String	1	Identifier for the reference document	
type_description	Document_Type	1	Description of the type of the Reference_Document	
language	String	0..1	The natural language used by the Reference_Document.	
title	String	0..1	The title of the Reference_Document	
provider	Organization	0..1	Organization that maintains or carries an official copy of the Reference_Document	
uri	String	0..1	Uri for Reference_Document	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
reference	Reference	0..*	References made to a Reference_Document	document_reference

4.3.20 Registrar

Registrar is a class each instance of which represents a registrar, a contact that is a representative of the registration authority. Registrars are the persons who perform the administrative steps to register administered items in a metadata registry.

Superclass: Contact

Attribute	**Datatype**	**Multiplicity**	**Description**	
identifier	String	1	The identifier of a Registrar	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
authority	Registration_Authority	1	Registration Authority for whom the Registrar administers metadata items.	registrar

4.3.21 Registration_Authority

Registration_Authority is a class each instance of which models a registration authority, an organization (i.e. a unique framework of authority within which a person or persons act, or are designated to act, towards some purpose) responsible for maintaining a register.

Superclass: Organization

Attribute	**Datatype**	**Multiplicity**	**Description**	
registration_authority_identifier	String	1	Identifier of a Registration_Authority. Organization identifiers within healthcare do not routinely conform to the identifier standard set out in MDRMetamodel and thus the strict rules for the composition of this identifier are relaxed. However the namespace within which the identifier is contained should be identified in fully compliant registries.	
name	String	1	Name of the Registration_Authority	
web_address	String	0..1	Web address of the Registration Authority	
standard	String	1	Standard to which the Registration_Authority conforms to	
conformance_level	String	1	The conformance level of the Registration_Authority to the nominated standard	
primary_language	String	0..1	The primary language of the Registration_Authority	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
administered_item	Administered_Item	0..*	Administered_Items registered by a Registration_Authority	authority
registrar	Registrar	1..*	Registrars who are conferred the authority to administer metadata items	authority
registration_namespace	Namespace	0..1	Namespace for the Registration_Authority	maintainer

4.3.22 Registration_Status

The designation of the level of registration, quality of metadata or progression of an Administered Item.

NOTE Registries might omit certain registration statuses as being inappropriate for a particular registration authority - a healthcare clinic is unlikely to develop its own preferred standards, but will record application level data elements and relate those to preferred standards maintained by other organizations.

Enumeration	Meaning
preferred_standard	The Registration Authority confirms that the Administered Item is preferred for use within the community that uses this metadata register.
standard	The Registration Authority confirms that the Administered Item is of sufficient quality and of broad interest for use in the community that uses this metadata register.
qualified	The Registration Authority has confirmed that the mandatory metadata attributes are complete and the mandatory metadata attributes conform to applicable quality
recorded	The Registration Authority has confirmed that all mandatory metadata attributes have been completed.
candidate	The Administered Item has been proposed for progression through the registration levels.
incomplete	Submitter wishes to make the community that uses this metadata register aware of the existence of an Administered Item in their local domain.
retired	The Registration Authority has approved the Administered Item as no longer recommended for use in the community that uses this metadata register and should no longer be used.
superseded	The Registration Authority determined that the Administered Item is no longer recommended for use by the community that uses this metadata register, and a successor Administered Item is now preferred for use.
historical	The Submitter wishes to make the community that uses this metadata register aware of the existence of an Administered Item that was used in the past.
application	The Registration Authority wishes to make the community that uses this metadata register aware of the existence of an Administered Item in their local domain that is in an application system and is not specified at the logical level. This item can be very well described.

4.3.23 Registry_Specification

Registry_Specification describes the environment in which a registry operates. If this document is being applied in an environment that does not use a registry, then Registry_Specification is not required. In an environment with multiple registries, there should be one Registry_Specification per registry.

Attribute	Datatype	Multiplicity	Description
name	String	1	Name of the registry
comment	String	0..1	Any comment about the registry
web_address	String	0..1	Web address for the registry
standard	String	0..1	The standard to which this registry conforms to
conformance_level	String	0..1	Conformance level of the registry to the standard specified
primary_language	String	0..1	Primary language of the registry
context	String	0..1	Name of the context of the registry, where the registry covers a single context. This attribute is mandatory if Designations and Definitions are not associated with a particular context.

4.3.24 Relation_Kind

The designation of the kind of relationship between Concepts in a Concept System.

Enumeration	Meaning
broader	Target concept is more general than the source concept
narrower	Target concept is more specific than the source concept
related	Source and target concepts are related

4.3.25 Role

Role is a class each instance of which models a role which an Individual might play as a Contact within an Organization

Attribute	Datatype	Multiplicity	Description
title	String	0..1	Name of the position that fills the Role
mail_address	Postal_Address	0..1	Postal address of the Role
email_address	String	0..*	Email address of the Role
phone_number	Phone_Number	0..*	Phone number of the Role

4.3.26 Scoped_Identifier

Scope Identifier is a class each instance of which models a scoped identifier, an identifier with a particular scope provided by a Namespace. For a description of the implementation of common healthcare metadata identification systems according to this document, see Annex B.

Attribute	Datatype	Multiplicity	Description
identifier	String	1	String used to unambiguously denote an Identified_Item within the scope of a specified Namespace
version	String	1	unique version identifier of the Scoped_Identifier which identifies and Identified_Item
full_expansion	String	0..1	String representation of a Scoped_Identifier in which thew unique identifier of the associated Namespace is combined in some way with the identifier of the Scoped_Identifier to fully specify the scope
shorthand_expansion	String	0..1	String representation of a Scoped_Identifier in which a shorthand_prefix from the associated Namespace has been prepended to the identifier to indicate the scope.

Reference	Class	Multiplicity	Description	Inverse
identified_item	Administered_Item	0..1	The Administered_Item which is unambiguously identified by a Scoped_Identifier in a Namespace	identifier
scope	Namespace	1	the Namespace for a Scoped_Identifier	contained_identifier

4.3.27 Semantic_Relation

Semantic Relation is a class each instance of which models a Concept related by meaning.

Attribute	Datatype	Multiplicity	Description
kind	Relation_Kind	0..1	Relationship kind between concepts

Reference	Class	Multiplicity	Description	Inverse
related_to	Concept	1	The Concept that is the target of a Semantic_Relation	relation
related_from	Concept	1	Source Concept for a Semantic_Relation	relation

4.3.28 Slot

Slot instances provide a dynamic way to add arbitrary attributes to instances of Administered_Item. A Slot instance is associated with exactly one Administered_Item instance, through the association item_slot. An Administered_Item instance can be associated with zero, one or more Slot instances. All Slot instances associated with a particular Administered_Item instance shall have a distinct name to allow each Slot instance to be unambiguously identified.

Attribute	Datatype	Multiplicity	Description	
name	String	1	Name of the Slot	
value	String	0..*	Value of the Slot	
type	String	0..1	Categorisation of the value of the Slot	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
item	Administered_Item	1	The Administered_Item for which a Slot extends	slot

4.4 Data Element Description

4.4.1 General

Subclause 4.4 describes the metadata to be recorded about a data element and a value domain – or value set – together with the conceptual framework which explains and identifies common meanings between registered data elements and their values(see Figure 3). This model is derived from MDRMetamodel, with extensions for valuesets defined by intension. For a more detailed comparison between this document and that of FHIR Value Sets, see Annex C. For a discussion of stringency and binding strength in value sets and how that relates to this document, see Annex D.

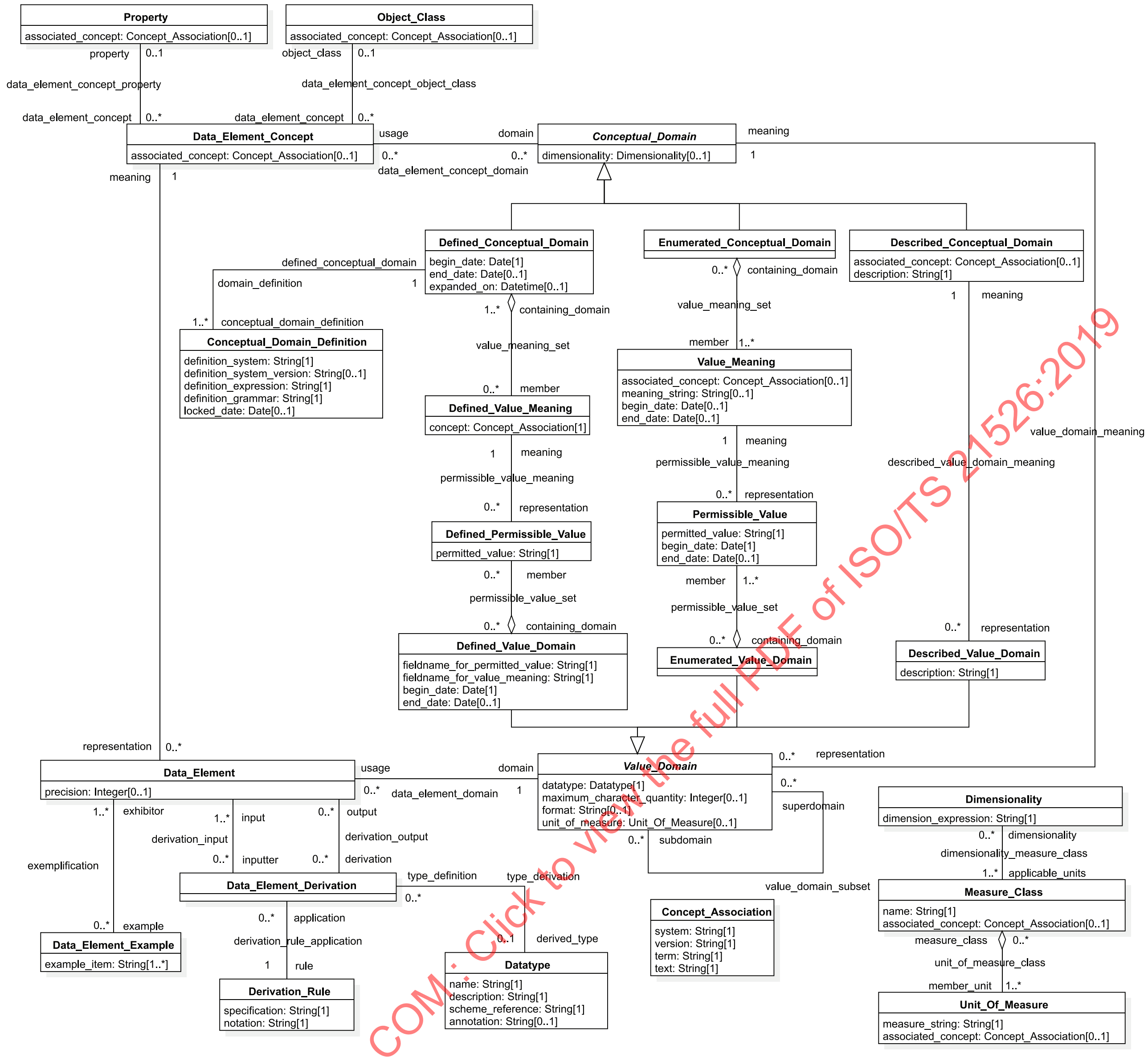

Figure 3 — Data Element Description

4.4.2 Association_Level

Association_Level is an enumeration that lists permissible values for the association_level attribute of Data_Element_Derivation.

Enumeration	Meaning
M0	Indicates that the Data_Element_Derivation is in regard to instance data - i.e. source data element values can be transformed into target element values by the application of the associated set of Derivation_Rule instances.
M1	Indicates that the Data_Element_Derivation is has regard to model data - i.e. source data elements can be combined to form a compound data element by the application of the associated set of Derivation_Rule instances.

4.4.3 Concept_Association

Concept_Association is a datatype, each instance specifies a term in an externally maintained concept system or terminology.

Attribute	Datatype	Multiplicity	Description
system	String	1	System referenced in the association EXAMPLE 1 http://snomed.info.
version	String	1	Version or release of the system referenced, or the version of the term referenced as appropriate
term	String	1	The term identifier referenced by the association EXAMPLE 2 404684003; 789-8; C18.2.
text	String	1	The textual representation of the term EXAMPLE 3 Clinical Finding (Finding); Erythrocytes [#/volume] in Blood by Automated count; Malignant neoplasm of ascending colon.

4.4.4 Conceptual_Domain

Conceptual_Domain is a class, each instance of which models a conceptual domain, a set of value meanings which can either be enumerated or expressed via a description. Conceptual_Domain is an abstract class which has three possible subclasses: Enumerated_Conceptual_Domain , Defined_Conceptual_Domain and Described Conceptual_Domain.

Superclass: Administered_Item

Attribute	Datatype	Multiplicity	Description
dimensionality	Dimensionality	0..1	Expression of measurement without units NOTE 1 When a dimensionality is specified, then any Unit_of_Measure specified for any Value_Domain that is based on this Conceptual_Domain, shall be consistent with this dimensionality. NOTE 2 ISO 80000-1 specifies physical dimensions (e.g. length, mass, velocity). This document also permits non-physical dimensions (e.g. value dimensions such as: currency, quality indicator).

Reference	Class	Multiplicity	Description	Inverse
representation	Value_Domain	0..*	Value domains providing representation for a conceptual domain	meaning
usage	Data_Element_Concept	0..*	Data element concepts included in a conceptual domain	domain

4.4.5 Conceptual_Domain_Definition

Conceptual_Domain_Definition is a class, each instance of which is an expression in a grammar that defines a set of Value_Meanings that are to be included in a Defined_Conceptual_Domain.

Where a Defined_Conceptual_Domain has more than one associated definition, then the full set of notions defined by the domain is the distinct union of the results of the expressions. If a term from the same terminology is selected by more than one definition expression, it shall appear in the resulting domain only once.

Attribute	Datatype	Multiplicity	Description
definition_system	String	1	The terminology system from which value meanings and permissible values are drawn
definition_ system_version	String	0..1	Version of the terminology system from which value meanings and permissible values are drawn
definition_expression	String	1	Expression in the specified definition_grammar that specifies the defined set of value meanings and permissible values within the definition_system

definition_grammar	String	1	Grammar for the definition_expression. EXAMPLE A dialect of regular expressions suitable for constraining LOINC value sets; SNOMED CT compositional grammar for constraining SNOMED CT value sets.
locked_date	Date	0..1	Date on which this definition became, or is to become locked in the absence of a defined version. NOTE Where source terminologies are under continual revision, where new terms are not automatically admitted into the Conceptual Domain, and where the terminology is not subject to a labelled release schedule, then a definition can be bound to terms defined before the locked date. See HL7 FHIR ValueSet resource.

Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
defined_conceptual_domain	Defined_Conceptual_Domain	1	The Defined_Conceptual_Domain specified in whole or in part by a Conceptual_Domain_Definition.	conceptual_domain_definition

4.4.6 Data_Element

Data_Element is a class each instance of which models a data element, a unit of data that is considered in context to be indivisible. A data element is a basic unit of data of interest to an organization, for which the definition, identification, representation, and permissible values are specified by means of a set of attributes. Examples of data element include: a column in a table of a relational database, a field in a record or form, an XML element, the attribute of a Java class, or a variable in a program.

Superclass: Administered_Item

Attribute	**Datatype**	**Multiplicity**	**Description**
precision	Integer	0..1	Number of decimal places permitted in any associated data element values

Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
meaning	Data_Element_Concept	1	The data element concept that encapsulates the notion of a data element, free of any constraints on representation	representation
domain	Value_Domain	1	The Value_Domain which provides representation for a Data_Element	usage
inputter	Data_Element_Derivation	0..*	Data_Element_Derivations that use a Data_Element	input
example	Data_Element_Example	0..*	Examples of a Data_Element	exhibitor
derivation	Data_Element_Derivation	0..*	Data_Element_Derivations for which a Data_Element provides a target	output

4.4.7 Data_Element_Concept

Data_Element_Concept is a class each instance of which models a data element concept. A data element concept is a specification of a concept independent of any particular representation. A data element concept can be represented in the form of a data element.

NOTE The combination of an object class and property provides significance beyond either that of the property or object class. A Data_Element_Concept thus has a definition independent of the definitions of the object class and property.

Superclass: Administered_Item

Attribute	**Datatype**	**Multiplicity**	**Description**
associated_concept	Concept_Association	0..1	An optional association with an externally maintained concept that corresponds to the data element concept in the metadata registry.

Reference	**Class**	**Multiplicity**	**Description**	**Inverse**

object_class	Object_Class	0..1	The Object_Class that is part of the specification of a Data_Element_Concept	data_element_concept
property	Property	0..1	The Property that is part of the specification of a Data_Element_Concept	data_element_concept
domain	Conceptual_Domain	0..*	Conceptual domains that encompass the notion of a data element concept	usage
representation	Data_Element	0..*	Data elements classified by a data element concept	meaning

4.4.8 Data_Element_Derivation

Data_Element_Derivation is a class each instance of which models a data element derivation, the application of a derivation rule to one or more input data elements to derive one or more output data elements, or to define a compound datatype.

A derivation shall have at least one data element or exactly one datatype as its target. Derivations that have a data element as their target encapsulate type conversion operations between the source and the target data elements - e.g. the conversion of a data element in °C to one in °F, or the construction of a pipe-delimited HL7v2 message string. Derivations that have a datatype as their target define a compound datatype - e.g. FHIR:Address; ISO 21090.

Superclass: Administered_Item

Reference	Class	Multiplicity	Description	Inverse
rule	Derivation_Rule	1	The rule for a Data_Element_Derivation	application
input	Data_Element	1..*	Data_Elements that provide input to a Derivation	inputter
output	Data_Element	0..*	Data_Elements that are outputs of a Data_Element_Derivation	derivation
derived_type	Datatype	0..*	The Datatype defined by a derivation	type_definition

4.4.9 Data_Element_Example

Data_Element_Example is a class each instance of which models a data element example, a representative illustration of a data element. Every Data_Element_Example shall have an exemplification association with one or more exhibitor Data_Elements, where the Data_Element_Example serves as an example for the Data_Element. A Data_Element_Example shall have one or more example_item attributes of type String that provide representative illustrations of instances of a Data_Element.

Attribute	Datatype	Multiplicity	Description
example_item	String	1..*	Actual illustrative case of the Data_Element

Reference	Class	Multiplicity	Description	Inverse
exhibitor	Data_Element	1..*	Exemplified data elements	example

4.4.10 Datatype

Datatype is a class, each instance of which models 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 might be performed on the Data_Element. Datatype is intended to accommodate datatypes from datatype schemes specified in external standards. It is also intended to accommodate other non-standard datatype schemes. Possible standardized datatype schemes include: the ISO/IEC 9075 series (SQL dataype); ISO/IEC 11404 on general purpose datatypes; ISO 21090 on Health Informatics datatypes; C programming language datatypes; XML Schema datatypes; etc.

Attribute	Datatype	Multiplicity	Description
name	String	1	Designation for the Datatype
description	String	1	Descriptive information to further clarify the Datatype
scheme_reference	String	1	Reference identifying the source of the Datatype specification
annotation	String	0..1	Specifying information to further define the Datatype

Reference	Class	Multiplicity	Description	Inverse
type_definition	Data_Element_Derivation	0..*	Definitions for this datatype.	derived_type

4.4.11 Defined_Conceptual_Domain

A Conceptual_Domain sometimes contains a finite allowed inventory of notions that can be defined logically. Such a Conceptual_Domain is referred to as a Defined_Conceptual_Domain.

EXAMPLE All descendants of SCTID: 443961001 Adenocarcinoma (disorder). As a subclass of Conceptual_Domain, a Defined_Conceptual_Domain inherits the attributes and relationships of the former.

Superclass: Conceptual_Domain

Attribute	Datatype	Multiplicity	Description
begin_date	Date	1	Date at which Value_Meanings defined within this Defined_Conceptual_Domain became, or will become valid Value_Meanings
end_date	Date	0..1	Date at which Value_Meanings defined within this Defined_Conceptual_Domain ceased, or will cease, to be valid. NOTE The absence of the value_meaning_end_date indicates that the Value_Meaning is still valid.
expanded_on	Datetime	0..1	Date on which the Defined_Conceptual_Domain was expanded, i.e. the set of rules in the Conceptual_Domain_Definition were played against the source terminology to produce a definitive set of Value_Meanings and Permissible_Values

Reference	Class	Multiplicity	Description	Inverse
member	Defined_Value_Meaning	0..*	Value meanings that have been expanded from the definition of a Defined_Conceptual_ Domain	containing_domain
conceptual_domain_ definition	Conceptual_Domain_ Definition	1..*	Definitions, the union of whose results are the set of notions that compose a Conceptual_Domain	defined_conceptual_ domain

4.4.12 Defined_Permissible_Value

Defined_Permissible_Value is a class each instance of which models a permissible value, the designation of a value meaning. A Permissible_Value is an expression of a Value_Meaning within zero or more Enumerated_Value_Domains.

NOTE 1 Defined_Permissible_Value instances are only present where a Defined_Conceptual_ Domain has been expanded.

NOTE 2 A Defined_Conceptual_Domain together with an expanded set of value meanings and permissible values might exist without association with a value domain to support the registration of HL7(FHIR) valuesets.

Attribute	**Datatype**	**Multiplicity**	**Description**	
permitted_value	String	1	Source permitted value from the concept system(s) included in the expansion	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
containing_domain	Defined_Value_Domain	0..*	Defined_Value_Domains that use a Defined_Permissible_Value	member
meaning	Defined_Value_Meaning	1	The Defined_Value_Meaning which provides meaning for a Defined_Permissible_Value	representation

4.4.13 Defined_Value_Domain

Defined_Value_Domain is a class each instance of which models a defined value domain defined by a function operating upon a suitable terminology which enumerates the list of its permissible values. The Described_Value_Domain class is a concrete subclass of the abstract class Value_Domain.

NOTE For Defined domains that have not been expanded, the value_domain_meaning association provides the link between Defined_Conceptual_Domain and Defined_Value_Domain.

Superclass: Value_Domain

Attribute	**Datatype**	**Multiplicity**	**Description**	
fieldname_for_permitted_value	String	1	Fieldname of an attribute of a controlled terminology specified by a Conceptual_Domain_Definition that provides the permitted value string used as the representation of its concepts.	
fieldname_for_value_meaning	String	1	Fieldname of an attribute of a controlled terminology specified by a Conceptual_Domain_Definition that provides the value meaning that is presented as the meaning of its concepts.	
begin_date	Date	1	Date at which the permissible values and value meanings defined by the Conceptual_Domain_Definition became valid	
end_date	Date	0..1	Date at which the permissible values and value meanings defined by the Conceptual_Domain_Definition ceased to be valid	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
member	Defined_Permissible_Value	0..*	Defined_Permissible_Values that provide representation for a Defined_Value_Domain	containing_domain

4.4.14 Defined_Value_Meaning

Defined_Value_Meaning is a class each instance of which models a value meaning that has been expanded from a Defined_Conceptual_Domain, which provides semantic content of a possible value.

Where Defined_Value_Meaning instances are present, the expanded_on attribute of Defined_Conceptual_Domain shall be set.

Attribute	**Datatype**	**Multiplicity**	**Description**	
concept	Concept_Association	1	Source concept from a concept system(s) included in the expansion.	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
representation	Defined_Permissible_Value	0..*	Defined_Permissible_Values which provide representation for a Defined_Value_Meaning	meaning
containing_domain	Defined_Conceptual_Domain	1..*	Defined_Conceptual_Domains of which the value meaning is a member	member

4.4.15 Derivation_Rule

Derivation_Rule is a class each instance of which models a derivation rule, logical, mathematical, and/or other operations specifying derivation. The Derivation_Rule can 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. A Derivation_Rule can have a derivation_rule_application association with zero or more application Data_Element_Derivations, where the Derivation_Rule provides the rule for the associated Data_Element_Derivation. A Derivation_Rule can be administered without necessarily being associated with any Data_Element_Derivation. As an Administered_Item, a Derivation_Rule is directly or indirectly associated with an Administration_Record and can be identified, named, defined and optionally classified as a Classifiable_Item in a Classification_Scheme.

Attribute	**Datatype**	**Multiplicity**	**Description**	
specification	String	1	Expression in the associated notation that executes the derivation	
notation	String	1	Notation of the associated expression.	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
application	Data_Element_Derivation	0..*	Data_Element_Derivations to which a Derivation_Rule applies	rule

4.4.16 Described_Conceptual_Domain

Described_Conceptual_Domain is a class each instance of which models a described conceptual domain, a conceptual domain that is specified by a description or specification, such as a rule, a procedure, or a range (i.e. interval), because it cannot be expressed as a finite set of value meanings. As a subclass of Conceptual_Domain, a Described Conceptual_Domain inherits the attributes and relationships of the former.

Superclass: Conceptual_Domain

Attribute	**Datatype**	**Multiplicity**	**Description**	
associated_concept	Concept_Association	0..1	Source concept from an externally maintained terminology associated with this Described_Conceptual_Domain. NOTE 1 It is anticipated that Described_Conceptual_Domain is administered in all implementations of this document. Where no concept is associated, the Described_Conceptual_Domain's identifier, name and description define it as a concept belonging to the default concept system of the registry.	
description	String	1	Prose description or specification of a rule, reference, or range for a set of all Value_Meanings for a Conceptual_Domain	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
representation	Described_Value_Domain	0..*	Described_Value_Domains providing representation for a Described_Conceptual_Domain.	meaning

4.4.17 Described_Value_Domain

Described_Value_Domain is a class each instance of which models a described value domain, a value domain that is specified by a description or specification, such as a rule, a procedure, or a range, rather than as an explicit set of permissible values. It is a concrete subclass of the abstract class Value_Domain. As a subclass of Value_Domain, a Described_Value_Domain inherits the attributes and relationships of the former.

Superclass: Value_Domain

Attribute	**Datatype**	**Multiplicity**	**Description**
description	String	1	Prose description or specification of a rule, reference, or range for a set of all Permissible_Values for a Value_Domain

Reference	Class	Multiplicity	Description	Inverse
meaning	Described_Conceptual_Domain	1	The Described_Conceptual_Domain which provides meaning for a Described_Value_Domain.	representation

4.4.18 Dimensionality

Dimensionality is a class each instance of which models a dimensionality, a specification in some reduced set of fundamental quantities for a unit of measure.

EXAMPLE 1 Inches, feet, metres, and centimetres are all units of measure whose dimensionality is length.

EXAMPLE 2 Mass, time, area, volume, etc. are other common dimensionalities.

NOTE Units of measure are not limited to physical categories. Examples of physical categories are linear measure, area, volume, mass, velocity, time duration. Examples of non-physical categories are currency, quality indicator, colour intensity.

Superclass: Administered_Item

Attribute	Datatype	Multiplicity	Description
dimension_expression	String	1	Expression of dimensionality. EXAMPLE 3 LT-1 is the dimensionality of velocity, which might be measured in m/s or miles per hour.

Reference	Class	Multiplicity	Description	Inverse
applicable_units	Measure_Class	1..*	Measure_Class instances which group related Unit_Of_Measure instances	dimensionality

4.4.19 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.

EXAMPLE The notion of countries that is specified in ISO 3166-1. As a subclass of Conceptual_Domain, an Enumerated_Conceptual_Domain inherits the attributes and relationships of the former.

Superclass: Conceptual Domain

Reference	Class	Multiplicity	Description	Inverse
member	Value_Meaning	1..*	Value_Meanings included within an Enumerated_Conceptual_Domain	containing_domain

4.4.20 Enumerated_Value_Domain

Enumerated_Value_Domain is a class each instance of which models an enumerated value domain, a value domain that is specified by a list of all its permissible values. The Enumerated_Value_Domain class is a concrete subclass of the abstract class Value_Domain.

Superclass: Value_Domain

Reference	Class	Multiplicity	Description	Inverse
member	Permissible_Value	1..*	Permissible_Values contained within an Enumerated_Value_Domain	containing_domain

4.4.21 Measure_Class

Measure_Class is a class each instance of which models a measure class, a set of equivalent units of measure that might be shared across multiple dimensionalities. Measure_Class allows a grouping of units of measure to be specified once, and reused by multiple dimensionalities.

EXAMPLE We could define the Measure_Classes: Metric Linear Distance, Imperial Linear Distance, each associated with the appropriate Units_of_Measure; and associate them with Dimensionalities: Height, Width, and Depth to model the three spatial dimensions.

Superclass: Administered_Item

Attribute	Datatype	Multiplicity	Description
name	String	1	Designation of the measure class
associated_concept	Concept_Association	0..1	Source concept from an externally maintained terminology associated with this Measure_Class

Reference	Class	Multiplicity	Description	Inverse
dimensionality	Dimensionality	0..*	Dimensionality instances which use Measure_Class to reference appropriate units of measure	applicable_units
member_unit	Unit_Of_Measure	1..*	Member units for a measure class	measure_class

4.4.22 Object_Class

Object_Class is a class each instance of which models an object class. An object class is a concept that represents 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 can be either a single or a group of associated concepts, abstractions, or things.

Superclass: Administered_Item

Attribute	Datatype	Multiplicity	Description
associated_concept	Concept_Association	0..1	An optional association with an externally maintained concept that corresponds to the object class in the metadata registry.

Reference	Class	Multiplicity	Description	Inverse
data_element_concept	Data_Element_Concept	0..*	Data_Element_Concepts that an Object_Class helps specify	object_class

4.4.23 Permissible_Value

Permissible_Value is a class each instance of which models a permissible value, the designation of a value meaning. A Permissible_Value is an expression of a Value_Meaning within zero or more Enumerated_Value_Domains.

Attribute	Datatype	Multiplicity	Description
permitted_value	String	1	The actual value of the Permissible_Value
begin_date	Date	1	Date at which the Permissible_Value became valid
end_date	Date	0..1	Date at which the Permissible_Value ceased to be valid

Reference	Class	Multiplicity	Description	Inverse
meaning	Value_Meaning	1	The Value_Meaning which provides meaning for a Permissible_Value	representation
containing_ domain	Enumerated_Value_Domain	0..*	Enumerated_Value_Domains containing a Permissible_Value	member

4.4.24 Property

Property is a class each instance of which models a property, a quality common to all members of an object class. A property can be any feature that humans naturally use to distinguish one individual object from another. It is the human perception of a single quality 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.

Superclass: Administered_Item

Attribute	**Datatype**	**Multiplicity**	**Description**
associated_concept	Concept_Association	0..1	An optional association with an externally maintained concept that corresponds to the property in the metadata registry.

Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
data_element_concept	Data_Element_Concept	0..*	Data_Element_Concepts that a Property helps to specify	property

4.4.25 Unit_Of_Measure

Unit_of_Measure is a class each instance of which models a unit of measure, the units in which associated values are measured. If appropriate, a Value_Domain can be associated with a Unit_of_Measure to specify the units in which any associated Data_Element values are measured.

Superclass: Administered_Item

Attribute	**Datatype**	**Multiplicity**	**Description**
measure_string	String	1	String designating the unit of measure EXAMPLE m/s; mMol/l.
associated_concept	Concept_Association	0..1	Source concept from an externally maintained terminology associated with this Unit_Of_Measure

Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
measure_class	Measure_Class	0..*	Measure classes that classify a unit of measure	member_unit

4.4.26 Value_Domain

Value_Domain is an abstract class each instance of which models a value domain, a set of permissible values. A value domain provides representation but has no implication as to the data element concept with which the values are associated, nor what the values mean. A Value_Domain has three possible subclasses: an Enumerated_Value_Domain, a Defined_Value_Domain and a Described_Value_Domain.

Superclass: Administered_Item

Attribute	**Datatype**	**Multiplicity**	**Description**
datatype	Datatype	1	Datatype used in a Value_Domain. NOTE 1 Applies to all values in the Value_Domain.
maximum_character_quantity	Integer	0..1	maximum number of characters available to represent the Data_Element value NOTE 2 Applicable only to character datatypes.
format	String	0..1	template for the structure of the presentation of the value(s) EXAMPLE YYYY-MM-DD for a date.

unit_of_measure	Unit_Of_Measure	0..1	unit of measure used in a Value_Domain NOTE 3 Applies to all values in the Value_Domain. NOTE 4 Constraints on unit of measure and dimensionality are specified in **.**.

Reference	Class	Multiplicity	Description	Inverse
superdomain	Value_Domain	0..*	Superdomains that contain a Value_Domain	subdomain
subdomain	Value_Domain	0..*	Subdomains that contain a Value_Domain	superdomain
usage	Data_Element	0..*	Data elements which use the representation described by a Value_Domain	domain
meaning	Conceptual_Domain	1	The conceptual domain for a value domain's value meanings	representation

4.4.27 Value_Meaning

Value_Meaning is a class each instance of which models a value meaning, which provides semantic content of a possible value. Each member of an Enumerated_Conceptual_Domain has a Value_Meaning that provides its distinction from other members. In the example of ISO 3166-1, 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 might have more than one means of representation by Permissible_Values — each from a distinct Enumerated_Value_Domain.

Attribute	Datatype	Multiplicity	Description
associated_ concept	Concept_Association	0..1	Source concept from an externally maintained terminology associated with this value meaning. A Value_Meaning shall have either an associated_concept or a meaning_string.
meaning_string	String	0..1	String representing the meaning of a permitted value. A Value_Meaning shall have either an associated_concept or a meaning_string.
begin_date	Date	0..1	Date at which this Value_Meaning became, or will become a valid Value_Meaning
end_date	Date	0..1	Date on which the Value_Meaning ceased, or will cease, to be valid. NOTE 1 The absence of the value_meaning_end_date indicates that the Value_Meaning is still valid.

Reference	Class	Multiplicity	Description	Inverse
containing_ domain	Enumerated_Conceptual_Domain	0..*	Enumerated_Conceptual_Domains for which a Value_Meaning is a member	member
representation	Permissible_Value	0..*	Permissible_Values which provide representation for a Value_Meaning	meaning

4.5 Data Model Description

4.5.1 General

Data model description (see Figure 4) provides a metamodel for the registration of tabular and class-based data models. This metamodel is derived from MFIInformationModel and includes only those classes, attributes and relations required for the recording of these kinds of models.

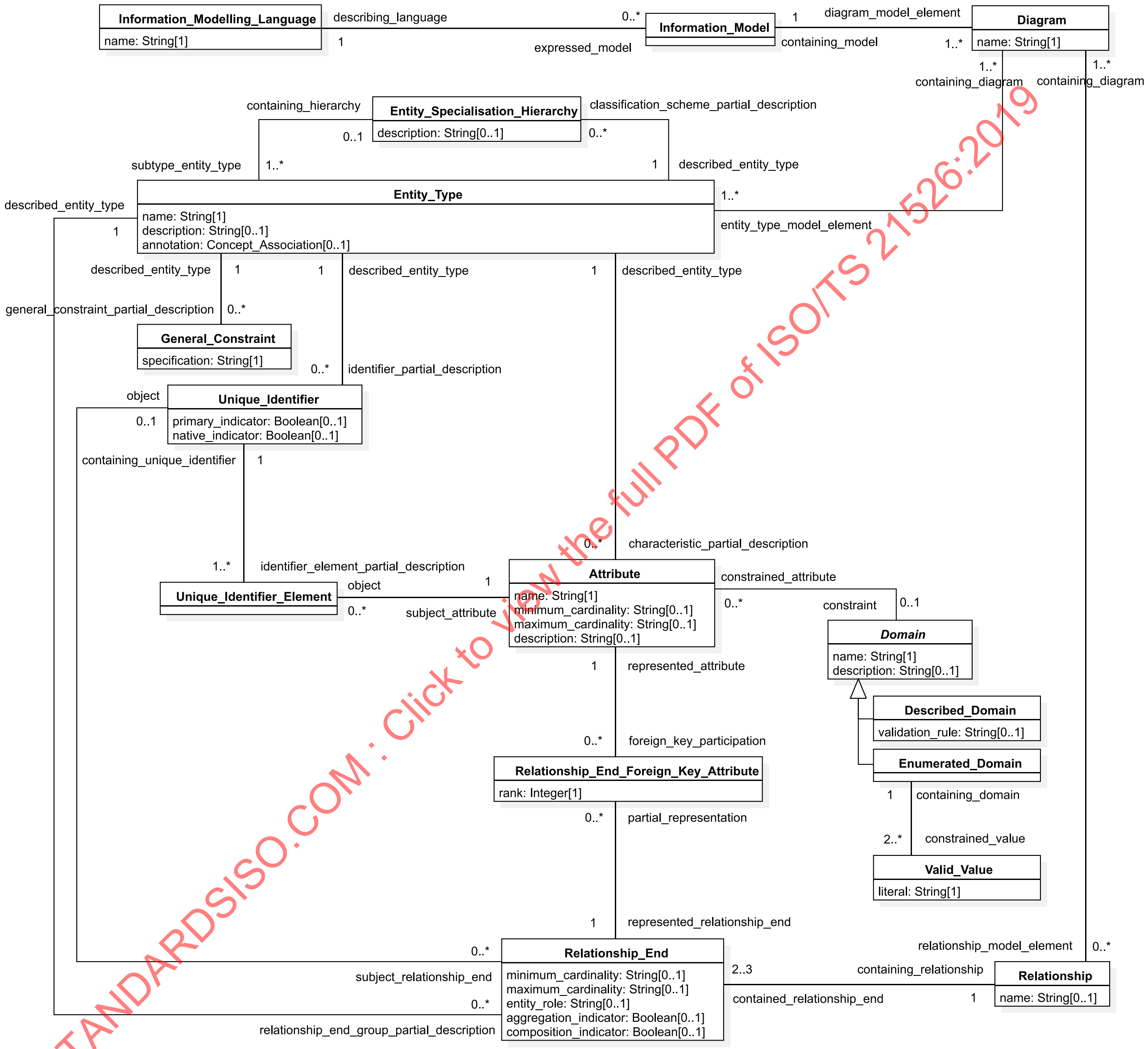
Information_Modelling_Language
name: String[1]
describing_language
1
0..*
expressed_model
Information_Model
1
diagram_model_element
containing_model
1..*
Diagram
name: String[1]
1..*
containing_diagram
1..*
containing_diagram
containing_hierarchy
Entity_Specialisation_Hierarchy
description: String[0..1]
0..1
classification_scheme_partial_description
0..*
subtype_entity_type
1..*
1
described_entity_type
Entity_Type
name: String[1]
description: String[0..1]
annotation: Concept_Association[0..1]
1..*
entity_type_model_element
described_entity_type
1
described_entity_type
1
1
described_entity_type
1
described_entity_type
general_constraint_partial_description
0..*
General_Constraint
specification: String[1]
0..*
identifier_partial_description
object
Unique_Identifier
0..1
primary_indicator: Boolean[0..1]
native_indicator: Boolean[0..1]
containing_unique_identifier
1
0..*
characteristic_partial_description
1..*
identifier_element_partial_description
Attribute
constrained_attribute
1
object
Unique_Identifier_Element
0..*
subject_attribute
name: String[1]
minimum_cardinality: String[0..1]
maximum_cardinality: String[0..1]
description: String[0..1]
0..*
constraint
0..1
Domain
name: String[1]
description: String[0..1]
1
represented_attribute
Described_Domain
validation_rule: String[0..1]
Enumerated_Domain
0..*
foreign_key_participation
Relationship_End_Foreign_Key_Attribute
rank: Integer[1]
1
containing_domain
0..*
partial_representation
2..*
constrained_value
Valid_Value
literal: String[1]
1
represented_relationship_end
relationship_model_element
0..*
0..*
Relationship_End
2..3
containing_relationship
Relationship
subject_relationship_end
minimum_cardinality: String[0..1]
maximum_cardinality: String[0..1]
entity_role: String[0..1]
aggregation_indicator: Boolean[0..1]
composition_indicator: Boolean[0..1]
contained_relationship_end
1
name: String[0..1]
0..*
relationship_end_group_partial_description

Figure 4 — Data Model Description

4.5.2 Attribute

Attribute is an abstract metaclass each instance of which represents a representation of a particular attribute, a column or an attribute in an information model.

Attribute instances that represent important notions shared between models should be related to an appropriate Data_Element instance whereas attribute instances that are simply associated with the functioning of instance systems might simply be described in place.

Attribute	Datatype	Multiplicity	Description
name	String	1	A name for this attribute.
minimum_ cardinality	String	0..1	A statement of the minimum number of occurrences of values of this attribute for any particular instance of the associated entity type. In most circumstances this will be '0' (indicating that the attribute is optional) or '1' (indicating that the attribute is mandatory).
maximum_cardinality	String	0..1	A statement of the maximum number of occurrences of values of this attribute for any particular instance of the associated entity type.
description	String	0..1	A statement that explains the significance of this attribute to the business and or organisation that is the subject of this information model.

Reference	Class	Multiplicity	Description	Inverse
described_entity_type	Entity_Type	1	The set of attributes that are used to qualify, identify, classify, quantify or express the state of any instance of this entity type.	characteristic_partial_ description
foreign_key_ participation	Relationship_End_ Foreign_Key_ Attribute	0..*	The set of attributes comprising this foreign key	represented_ attribute
constraint	Domain	0..1	The set of attributes to which this domain applies as a constraint.	constrained_ attribute
object	Unique_Identifier_ Element	0..*	The set of attributes that are used as this unique identifier element.	subject_attribute

4.5.3 Described_Domain

Described_Domain is a metaclass each instance of which represents a representation of a particular described domain.

Superclass: Domain

Attribute	Datatype	Multiplicity	Description
validation_rule	String	0..1	A statement of the validation that might be applied to this domain such as an aspirational datatype; a permitted range; or a regular expression.

4.5.4 Diagram

Diagram is a metaclass each instance of which represents a representation of a particular diagram.

If an information model contains a single diagram, then this diagram shall contain all of the entities and relationships in the information model.

Attribute	Datatype	Multiplicity	Description
name	String	1	The name by which a diagram is known.

Reference	Class	Multiplicity	Description	Inverse
containing_model	Information_Model	1	The set of diagrams which comprise this information model.	diagram_ model_element
entity_type_ model_element	Entity_Type	1..*	The diagram which includes this entity type	containing_ diagram
relationship_ model_element	Relationship	0..*	The set of relationships contained within this information diagram.	containing_ diagram

4.5.5 Domain

Domain is an abstract metaclass each instance of which represents a representation of a particular domain.

Domain instances that represent important notions shared between models should be related to an appropriate Value_Domain instance whereas attribute instances that are simply associated with the functioning of instance systems might simply be described in place.

Attribute	Datatype	Multiplicity	Description
name	String	1	A name by which this domain is known.
description	String	0..1	A statement that explains the significance of this domain to the business and or organisation that is the subject of this information model.

Reference	Class	Multiplicity	Description	Inverse
constrained_attribute	Attribute	0..*	The domain which acts as a constraint on the values taken by this attribute.	constraint

4.5.6 Entity_Specialisation_Hierarchy

Entity_Specialisation_Hierarchy is a metaclass each instance of which represents a representation of a particular entity specialisation hierarchy.

Attribute	Datatype	Multiplicity	Description
description	String	0..1	A statement that describes the purpose or the classification of this particular entity specialisation hierarchy.

Reference	Class	Multiplicity	Description	Inverse
described_entity_type	Entity_Type	1	The set of hierarchies to which this entity optionally belongs.	classification_scheme_partial_description
containing_hierarchy	Entity_Specialisation_Hierarchy	0..1	The set of entity types that participate as subtypes in this entity specialisation hierarchy	subtype_entity_type

4.5.7 Entity_Type

Entity_Type is a metaclass each instance of which represents a representation of a particular entity type, a table or a class in an information model.

Attribute	Datatype	Multiplicity	Description
name	String	1	A name by which this entity type is known.
description	String	0..1	A statement that explains the significance of this entity type to the business and or organisation that is the subject of this Information Model.
annotation	Concept_Association	0..1	A statement that globally identifies a concept in a domain ontology that expresses the meaning or scope of this entity type.

Reference	Class	Multiplicity	Description	Inverse
containing_diagram	Diagram	1..*	The diagram which includes this entity type.	entity_type_model_element
containing_ hierarchy	Entity_Type	1..*	The set of entity specialisation hierarchies which include this entity type as a subtype.	subtype_entity_type
general_constraint_partial_description	General_Constraint	0..*	A set of constraints defined for this entity type.	described_entity_type

relationship_end_group_partial_description	Relationship_End	0..*	The set of relationship ends for an entity type.	described_entity_type
characteristic_partial_description	Attribute	0..*	The set of attributes that are used to qualify, identify, classify, quantify or express the state of any instance of this entity type.	described_entity_type
identifier_partial_description	Unique_Identifier	0..*	Unique identifiers defined for this entity type.	described_entity_type
classification_scheme_partial_description	Entity_Specialisation_Hierarchy	0..*	The set of hierarchies to which this entity optionally belongs.	described_entity_type

4.5.8 Enumerated_Domain

Enumerated_Domain is a metaclass each instance of which represents a particular enumerated domain.

Superclass: Domain

Reference	Class	Multiplicity	Description	Inverse
constrained_value	Valid_Value	2..*	The set of valid values that comprise this discrete domain.	containing_domain

4.5.9 General_Constraint

General_Constraint is a metaclass each instance of which is a representation of a particular general constraint.

Attribute	Datatype	Multiplicity	Description	
specification	String	1	A statement of that formally specifies this constraint.	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
described_entity_type	Entity_Type	1	The entity type that is constrained by this constraint.	general_constraint_partial_description

4.5.10 Information_Model

Information_Model is a metaclass each instance of which represents a particular information model.

Information_Models shall be Administered Items and thus are named, defined, identified, classified and administered in the same way as data elements and value domains.

Reference	Class	Multiplicity	Description	Inverse
describing_language	Information_Modelling_Language	1	Information modelling language in which this model is expressed	expressed_model
diagram_model_element	Diagram	1..*	The information model of which this diagram is a part	containing_model

4.5.11 Information_Modelling_Language

Information_Modelling_Language is a metaclass each instance of which represents a particular information modelling language.

Attribute	Datatype	Multiplicity	Description	
name	String	1	A name by which this information modelling language is known.	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**

expressed_model	Information_Model	0..*	Set of information models that are expressed in this language.	describing_language

4.5.12 Relationship

Relationship is a metaclass each instance of which represents a particular relationship.

Attribute	Datatype	Multiplicity	Description
name	String	0..1	A name by which this relationship is known. Some information modelling methods do not provide such a name.

Reference	Class	Multiplicity	Description	Inverse
containing_diagram	Diagram	1..*	The set of diagrams which include this relationship.	relationship_model_element
contained_relationship_end	Relationship_End	2..3	The set of relationship ends that comprise this relationship.	containing_relationship

4.5.13 Relationship_End

Relationship_End is a metaclass each instance of which represents a particular relationship end.

Attribute	Datatype	Multiplicity	Description
minimum_cardinality	String	0..1	A statement of the maximum number of instances of the associated entity type (through the associated relationship end group) that participate in the relationship of which this relationship end is a part. In most circumstances this will be '1' (indicating that one and only one entity type participates) or '*' (indicating that an unspecified number of entity types participate).
maximum_cardinality	String	0..1	A statement of the maximum number of instances of the associated entity type (through the associated relationship end group) that can participate in the relationship of which this relationship end is a part. In most circumstances this will be '1' (indicating that one and only one entity type can participate) or '*' (indicating that an unspecified number of entity types can participate).
entity_role	String	0..1	A statement that explains the role that the associated entity type (through the associated relationship end group) is playing in the associated relationship. Not all information modelling methods recognize this concept.
aggregation_indicator	Boolean	0..1	An indicator that specifies whether the instance of the associated entity type (through the associated relationship end group) is considered to be an aggregation of the instances of the other entity type participating in the relationship (identified through the associated relationship and relationship end group) or not. Most information modelling methods do not recognize this concept.
composition_indicator	Boolean	0..1	An indicator that specifies whether the instance of the associated entity type (through the associated relationship end group) is considered to be a composition of the instances of the other entity type participating in the relationship (identified through the associated relationship and relationship end group) or not. Most information modelling methods do not recognize this concept.

Reference	Class	Multiplicity	Description	Inverse
containing_relationship	Relationship	1	The relationship of which this relationship end is a part.	contained_relationship_end
object	Unique_Identifier	0..1	The set of relationships in which this unique identifier participates.	subject_relationship_end
described_entity_type	Entity_Type	1	The set of relationship ends for an entity type.	relationship_end_group_partial_description

partial_representation	Relationship_End_Foreign_Key_Attribute	0..*	The set of foreign key attributes that in sequence represent this relationship end.	represented_relationship_end

4.5.14 Relationship_End_Foreign_Key_Attribute

Relationship_End_Foreign_Key_Attribute is a metaclass each instance of which represents a particular relationship end foreign key attribute.

Attribute	Datatype	Multiplicity	Description
rank	Integer	1	A statement of the position of this foreign key attribute in the foreign key of which it is a part.

Reference	Class	Multiplicity	Description	Inverse
represented_relationship_end	Relationship_End	1	The relationship end which is, along with others in sequence, represented by this foreign key attribute.	partial_representation
represented_attribute	Attribute	1	The set of attributes comprising this foreign key.	foreign_key_participation

4.5.15 Unique_Identifier

Unique_Identifier is a metaclass each instance of which represents a particular unique identifier.

Attribute	Datatype	Multiplicity	Description
primary_indicator	Boolean	0..1	An indicator that specifies whether this unique identifier is the primary unique identifier of the associated entity or not. Some information modelling methods do not recognize this concept.
native_indicator	Boolean	0..1	An indicator that specifies whether this unique identifier is native key of the associated entity or not. Some information modelling methods do not recognize this concept.

Reference	Class	Multiplicity	Description	Inverse
described_entity_type	Entity_Type	1	The entity type whose instances can be uniquely identified by this unique identifier.	identifier_ partial_description
subject_relationship_end	Relationship_End	0..*	The set of relationship ends in which this unique identifier participates.	object
identifier_element_partial_description	Unique_Identifier_Element	1..*	The set of unique identifier elements that comprise this unique identifier.	containing_ unique_identifier

4.5.16 Unique_Identifier_Element

Unique_Identifier_Element is an abstract metaclass each instance of which represents a representation of a particular unique identifier element.

Reference	Class	Multiplicity	Description	Inverse
containing_unique_identifier	Unique_Identifier	1	The unique identifier of which this unique identifier element is a part.	identifier_ element_partial_ description

subject_attribute	Attribute	1	The set of attributes that are used as this unique identifier element.	object

4.5.17 Valid_Value

Valid_Value is a metaclass each instance of which represents a representation of a particular valid value.

Attribute	Datatype	Multiplicity	Description
literal	String	1	The actual permitted value. In a platform independent model it will probably be the concept, for example, 'Male'. In a platform specific model it will probably be the code, for example, 'M' or '0'.

Reference	Class	Multiplicity	Description	Inverse
containing_domain	Enumerated_Domain	1	The domain to which this valid value belongs.	constrained_value

4.6 XML Schema

4.6.1 General

Subclause 4.6 describes a metamodel for the registration of XML Schemas (see Figure 5) and for the identification of features within these models for association with data elements, value sets, and value meanings. MFIInformation model does not yet have a part to cover the recording of XML Schema models; this metamodel supports the minimum requirement for registration and association of model components with other kinds of registry content.

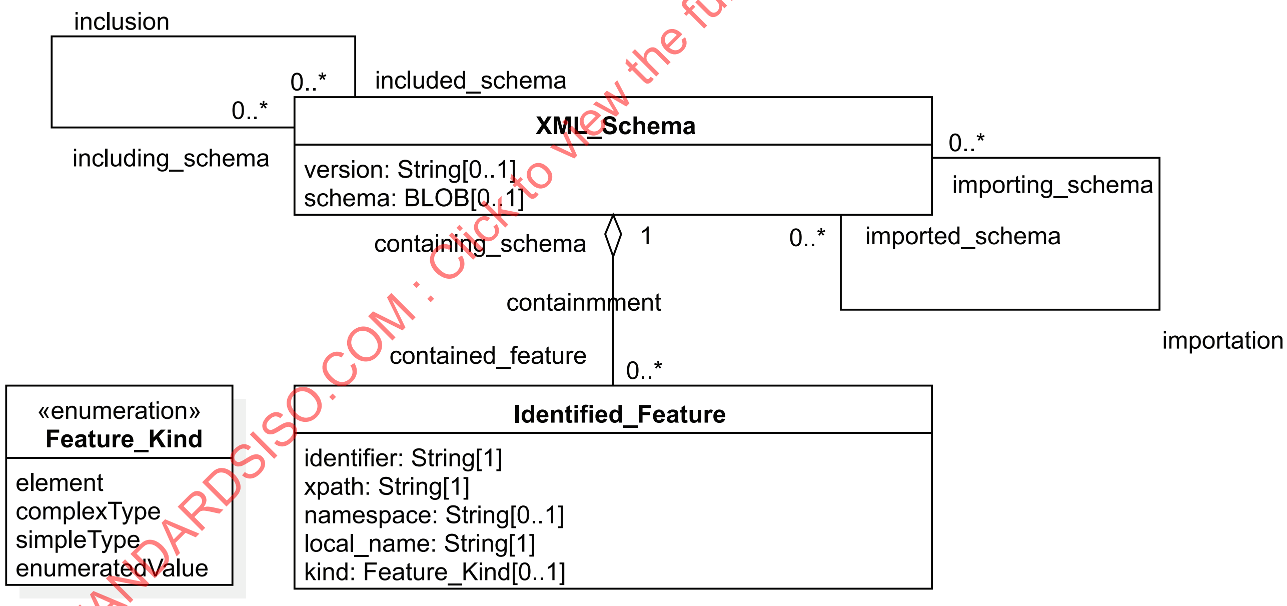

Figure 5 — XML Schema description

4.6.2 Feature_Kind

Feature_Kind is an enumeration of Identified_Feature kinds in an XML_Schema.

Enumeration	Meaning
complexType	Feature is a complex type.
simpleType	Feature is a simple type.
enumeratedValue	Feature is an xsd:enumeration within a simple type.

4.6.3 Identified_Feature

Identified_Feature is a metaclass each instance of which represents a feature or set of features - an element, complexType, simpleType or enumerated value - that has been located by an xpath statement and assigned an identifier.

NOTE Identified_Feature provides a stub for association with other registry content through the mapping mechanism.

Attribute	Datatype	Multiplicity	Description	
identifier	String	1	Identifier for the feature.	
xpath	String	1	Xpath expression that locates the feature within the schema	
namespace	String	0..1	Namespace of the feature	
local_name	String	1	Local name of the feature	
kind	Feature_Kind	0..1	Kind of feature located	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
containing_schema	XML_Schema	1	The Schema containing an Identified_Feature	contained_feature

4.6.4 XML_Schema

XML_Schema is a metaclass, each instance of which represents a W3C XML Schema.

Superclass: Administered_Item

Attribute	Datatype	Multiplicity	Description	
version	String	0..1	XML schema version for the schema	
schema	BLOB	0..1	The schema itself	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
included_schema	XML_Schema	0..*	Schemas that are included in this schema	including_schema
importing_schema	XML_Schema	0..*	Schemas that are imported by this schema	imported_schema
contained_feature	Identified_Feature	0..*	Identified_Features contained within a schema	containing_schema

4.7 Mapping

4.7.1 General

Mapping (see Figure 6) provides a metamodel for recording typed associations between different kinds of administered items.

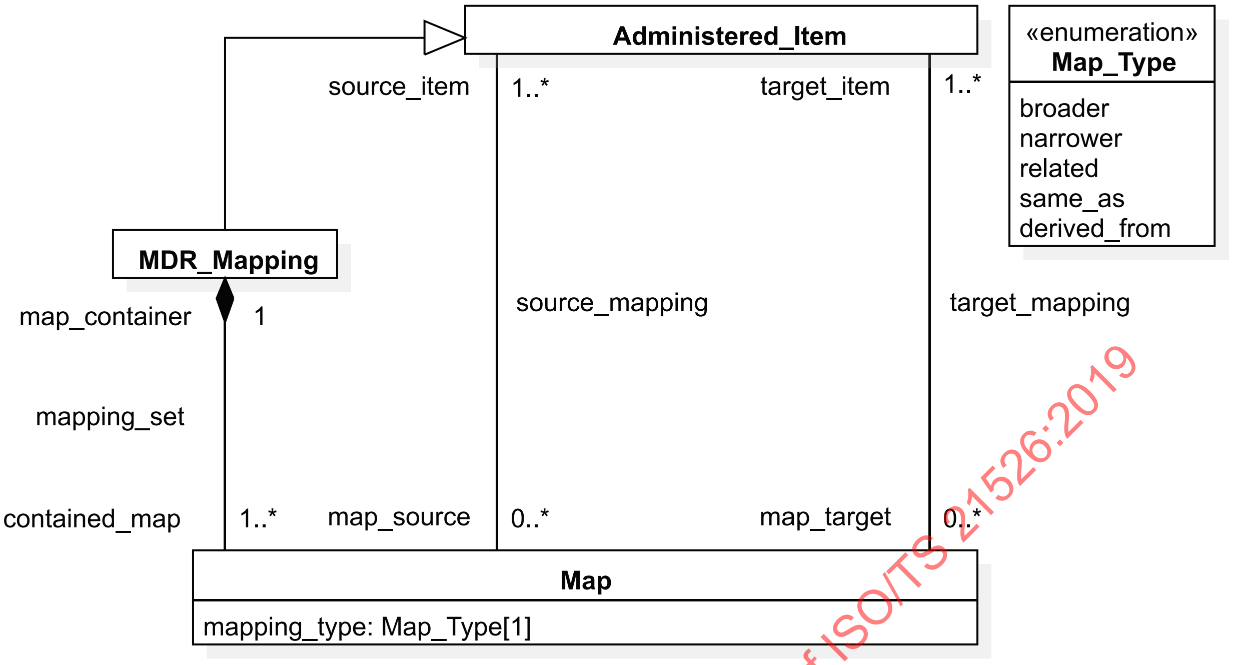

Figure 6 — Mapping

4.7.2 MDR_Mapping

MDR_Mapping is a metaclass each instance of which represents a set of Maps between administered items that are all simultaneously asserted to be valid

Superclass: Administered_Item

Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
contained_map	Map	1..*	Map instances contained within a MDR_Mapping	map_container

4.7.3 Map

Map is a metaclass each instance of which represents a map between sets of administered items.

Attribute	**Datatype**	**Multiplicity**	**Description**	
mapping_type	Map_Type	1	Semantics for the Map.	
Reference	**Class**	**Multiplicity**	**Description**	**Inverse**
map_container	MDR_Mapping	1	The MDR_Mapping that groups a set of related Maps	contained_map
source_item	Administered_Item	1..*	Administered items involved in this Map	map_source
target_item	Administered_Item	1..*	References one to many administered items that are the target of a Map	map_target

4.7.4 Map_Type

Map_Type is an enumerated type, each enumeration describes a kind of relationship between administered items.

Implementers shall include the Map_Type enumerations described in this document. Extensions are permitted provided they are subtypes of one of the standard enumerations.

Enumeration	Meaning
broader	The target administered item is more general than the source
narrower	The target administered item is more specific than the source
related	The target administered item might be of interest to a user accessing the source
same_as	The target administered item is exactly the same in definition as the source
derived_from	The target administered item was created from the source