INTERNATIONAL STANDARD

ISO/IEC 11179-3

Third edition
2013-02-15

AMENDMENT 1
2020-01

Information technology — Metadata registries (MDR) —

Part 3: Registry metamodel and basic attributes

AMENDMENT 1

Technologies de l'information — Registres de métadonnées (RM) —
Partie 3: Métamodèle de registre et attributs de base
AMENDEMENT 1

Reference number
ISO/IEC 11179-3:2013/Amd.1:2020(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

Foreword

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

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 document 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 and IEC 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) or the IEC list of patent declarations received (see http://patents.iec.ch).

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 Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 32, *Data management and interchange*.

A list of all parts in the ISO/IEC 11179 series can be found on the ISO website.

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.

Information technology — Metadata registries (MDR) —

Part 3:
Registry metamodel and basic attributes

AMENDMENT 1

3.1, 3.2

Change all references to "*datatype* (3.1.8)" to "*datatype* (3.1.9)".

3.1.13

Change the reference in the source from "7.3.37" to "7.3.38".

3.2.9

Replace the definition with:

attribute (3.1.4) of a *metadata item* (3.2.75) commonly needed in its specification

3.2.50

Replace the definition with:

metadata item (3.2.75) which can have *designations* (3.2.51) and/or *definitions* (3.2.40)

3.2.75

In NOTE 1, replace 'Clauses 5 through 11' with 'Clauses 9 through 11'.

3.2.76

In the NOTE, replace 'Clauses 5 through 11' with 'Clauses 9 through 11'.

3.2.96

Replace the existing NOTE by the following:

Note 1 to entry: Permissible values may be specified either as part of a *value domain* (3.2.140), or only associated with a *value meaning* (3.2.141).

Note 2 to entry: Within a value domain, permissible values may either be enumerated or described.

Note 3 to entry: Explicit mapping of a single permissible value to a single value meaning is possible only when both the value meaning and permissible value are enumerated, e.g. for code sets. For described permissible values, the described meaning might be associated with a range of values, e.g. weight in kilograms.

3.2.105

Replace the definition with the following:

identified_item (3.2.64) that is recorded and managed in a *metadata registry* (3.2.78)

3.2.138

In NOTE 2, replace 'must' with 'shall'.

3.2

Add the following new terms and definitions:

3.2.143
conceptual domain definition
formal definition of a *defined conceptual domain* (3.2.144)

Note 1 to entry: The definition may reference externally enumerated value meanings.

3.2.144
defined conceptual domain
conceptual domain (3.2.21) that is specified by a formal definition

Note 1 to entry: The definition may reference externally enumerated value meanings.

3.2.145
defined value domain
value domain (3.2.140) that is specified by reference to an external specification

Note 1 to entry: The external specification should enumerate the permissible values.

3.3.24

Replace 'XTML' with 'XTM'.

Clauses 4 through 12

Replace all occurrences of 'must' in normative text with 'shall', except for:

— 7.3.2.7.2.4 where 'must' should simply be deleted;

— 9.1.2.5.3.1 where 'must' is correct;

— 11.1.1 replace 'must' with 'should';

— 11.4.2.3.2 where 'must' is correct;

— Annexes C, D and F, which are informative.

4.2.2

Replace the whole subclause with the following:

A strictly conforming implementation:

a) shall support all mandatory, optional and conditional classes, attributes and associations;

b) shall not use, test, access, or probe for any extension features nor extensions to classes, attributes and/or associations;

c) shall not recognize, nor act on, nor allow the production of classes, attributes and/or associations that are dependent on any unspecified, undefined, or implementation-defined behaviour.

NOTE The use of extensions to the metamodel or the basic attributes can cause undefined behaviour.

4.3

Rename 4.3 to 'Conformance by feature'.

Append:

", or to specific features within these clauses"

at the end of the first sentence of 4.3.

4.4.2

Add the following NOTE at the end of 4.4.2:

NOTE The additional provisions specified in the above profiles apply only to these profiles, and not to conformance by feature in 4.3.

5.4

Replace the sentence:

"Clauses 6 through 11 of this part of ISO/IEC 11179 specify the types of *metadata objects* that form the structure of a *metadata registry*."

With:

"Clauses 6 through 8 of this part of ISO/IEC 11179 specify common facilities that form the core of a *metadata registry* and which apply to potentially any type of registry content. Clauses 9 through 11 of this part of ISO/IEC 11179 specify the types of *metadata objects* that form the potential content of a metadata registry."

5.5.1, list item 2, NOTE

Replace the reference to ISO/IEC 19763-2 with ISO/IEC 19763-10.

Clause 6 (except 6.3.3.1 and 6.3.5.1, see below), and 7.3.2.1:

Change all references to "*datatype* (3.1.8)" to " *datatype* (3.1.9)".

6.3.3.1 and 6.3.5.1

Change all references to "composite *datatype* (3.1.8)" to "*composite datatype* (3.1.7)".

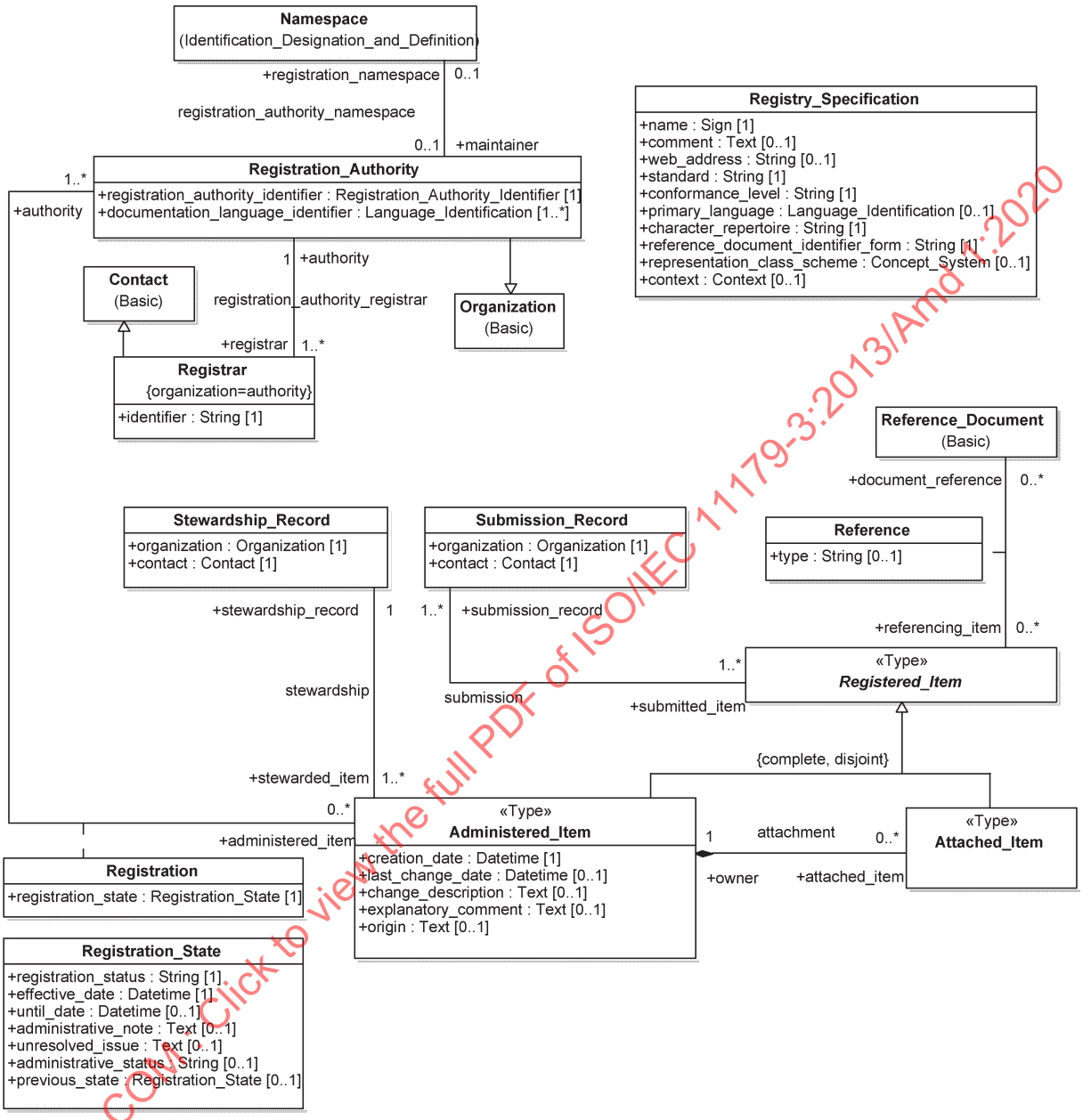
8.1.1, paragraph 1

Replace the second sentence with the following:

ISO/IEC 11179-6 further describes the registration of *Registered_Items* (8.1.2.1), *Administered_Items* (8.1.2.2) and *Attached_Items* (8.1.2.3).

8.1.1, Figure 7

Replace Figure 7 with the following:



8.1.2.9.1, list item 1, and Annex A, name (attribute of Registry_Specification)

Replace the reference to 8.1.2.9.2.1.1 with 8.1.2.9.2.1.

8.1.2.9.2

Change the subclause number of 8.1.2.9.2.1.1 to 8.1.2.9.2.1.

8.1.5.1.1, paragraph 1

Insert the following after the first sentence:

The association has two roles:

— authority (verb form: has_authority) which references a *Registration_Authority* class;

— administered_item (verb form: for_administered_item) which references an *Administered_Item* class.

An authority may have zero or more administered_items. An administered_item shall have one or more authorities.

8.1.5.2.1, paragraph 1

Insert the following after the first sentence:

The association has two roles:

— document_reference (verb form: has_document_reference) which references a *Reference_Document* class;

— referencing_item (verb form: is_referencing_item) which references a *Registered_Item* class.

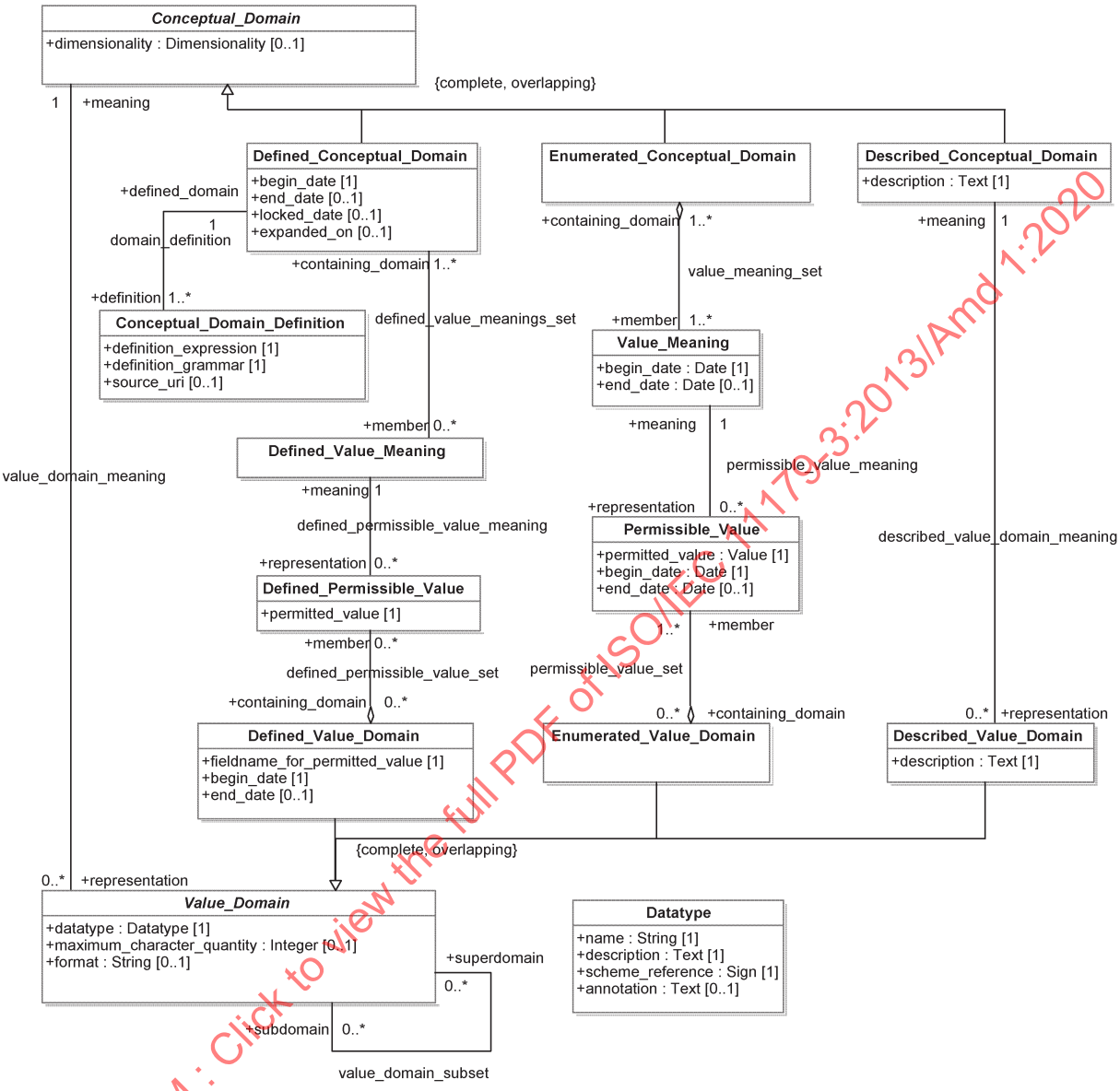
A document_reference may have zero or more referencing_items. A referencing_item may have zero or more document_references.

11.2.2.3.2, paragraph 1

In the last sentence, replace "in the form of" with "by".

11.3.1, Figure 13

Replace Figure 13 with the following:



11.3.2.5.2.3

Designate the existing NOTE as NOTE 1, and add the following NOTE:

NOTE 2 *maximum_character_quantity* is intended to be specified primarily for described value domains. If specified for enumerated_value_domains, care should be taken to ensure the specified value is consistent with the enumerated permissible values.

11.3.2.9

At the end of 11.3.2.9, insert subclauses 11.3.2.10 through 11.3.2.14:

11.3.2.10 Conceptual_Domain_Definition class

11.3.2.10.1 Description of Conceptual_Domain_Definition

Conceptual_Domain_Definition is a class, each instance of which models a *conceptual domain definition* (3.2.144), a formal definition of a *defined conceptual domain* (3.2.145).

Conceptual_Domain_Definition shall participate in a *domain_definition* association (11.3.3.7) with exactly one *Defined_Conceptual_Domain* (11.3.2.11).

11.3.2.10.2 Attributes of Conceptual_Domain_Definition

11.3.2.10.2.1 definition_expression

Attribute name: *definition_expression*

Definition: formal definition of the conceptual_domain expressed in some formal grammar

Obligation: Mandatory

Multiplicity: *1*

Datatype: *Text* (6.2.12)

11.3.2.10.2.2 definition_grammar

Attribute name: *definition_grammar*

Definition: name of the grammar used in *definition_expression*

Obligation: Mandatory

Multiplicity: *1*

Datatype: *String* (6.2.11)

11.3.2.10.2.3 source_uri

Attribute name: *source_uri*

Definition: Uri for the source of the *Conceptual_Domain_Definition*

Obligation: Optional

Multiplicity: *0..1*

Datatype: *String* (6.2.11)

—— End of attributes of *Conceptual_Domain_Definition* ——

11.3.2.11 Defined_Conceptual_Domain class

11.3.2.11.1 Direct superclass

Conceptual_Domain (11.3.2.1).

11.3.2.11.2 Description of Defined_Conceptual_Domain

Defined_Conceptual_Domain is a class, each instance of which models a *defined conceptual domain* (3.2.145).

As a subclass of *Conceptual_Domain, Defined_Conceptual_Domain* inherits the associations of the former.

In addition, *Defined_Conceptual_Domain* shall participate in a *domain_definition* association with one or more *Conceptual_Domain_Definitions*.

Where multiple *Conceptual_Domain_Definitions* are specified, the *definition_expression* (11.3.2.10.2.1) of each shall be equivalent, each specified using a different *definition_grammar* (11.3.2.10.2.2).

11.3.2.11.3 Attributes of Defined_Conceptual_Domain

11.3.2.11.3.1 begin_date

Attribute name: *begin_date*

Definition: date on which this *Defined_Conceptual_Domain* became, or will become, a valid *Defined_Conceptual_Domain*

Obligation: Mandatory

Multiplicity: *1*

Datatype: *Date* (6.2.3)

NOTE A *registration authority* can determine whether this date is the date the *defined conceptual domain* becomes valid in a registry, or the date the *defined conceptual domain* becomes part of the source domain, or some other date.

11.3.2.11.3.2 end_date

Attribute name: *end_date*

Definition: date on which the *Defined_Conceptual_Domain* ceased, or will cease, to be valid.

Obligation: Optional

Multiplicity: *0..1*

Datatype: *Date* (6.2.3)

NOTE 1 The absence of the *end_date* indicates that the *Defined_Conceptual_Domain* is still valid.

NOTE 2 A *registration authority* can determine whether this date is the date the *defined conceptual domain* becomes no longer valid in a registry, or the date the *defined conceptual domain* becomes no longer part of the source domain, or some other date.

11.3.2.11.3.3 locked_date

Attribute name: *locked_date*

Definition: date on which the set of *Defined_Value_Meanings* associated with the *Defined_Conceptual_Domain* was, or will be, locked.

Obligation: Optional

Multiplicity: *0..1*

Datatype: *Date* (6.2.3)

11.3.2.11.3.4 expanded_on

Attribute name: expanded_on

Definition: datetime at which the set of *Defined_Value_Meanings* associated with the *Defined_Conceptual_Domain* was expanded and imported into the registry.

Obligation: Optional

Multiplicity: *0..1*

Datatype: *Datetime* (6.2.4)

—— End of attributes of *Defined_Conceptual_Domain* ——

11.3.2.12 Defined_Value_Meaning class

11.3.2.12.1 Direct superclass

Defined_Value_Meaning is a subclass of *Concept* (6.4.2.2, 9.1.2.1).

11.3.2.12.2 Description of Defined_Value_Meaning

Defined_Value_Meaning is a class, each instance of which models a **value meaning** (3.2.142), which provides semantic content of a possible value.

NOTE *Defined_Value_Meaning* is like *Value_Meaning* (11.3.2.8), except importing the value meanings for a *Defined_Conceptual_Domain* is optional, and if done will be done as a set, so the *begin_date* and *end_date* attributes reside on the domain instead of the individual values.

Defined_Value_Meaning shall participate in a *defined_value_meaning_set* association (11.3.3.8) with one or more *Defined_Conceptual_Domains* (11.3.2.11).

Defined_Value_Meaning may participate in a *defined_permissible_value_meaning* (11.3.3.9) association with zero or more *Defined_Permissible_Values* (11.3.2.14).

11.3.2.12.3 Attributes of Defined_Value_Meaning

None.

11.3.2.13 Defined_Value_Domain class

11.3.2.13.1 Direct superclass

Value_Domain (11.3.2.8)

11.3.2.13.2 Description of Defined_Value_Domain

Defined_Value_Domain is a class each instance of which models a *defined value domain* (3.2.145), a *value domain* (3.2.140) that is specified by a reference to an external source of all its *permissible values* (3.2.96). The external source is specified by the associated the *Defined_Conceptual_Domain* (11.3.2.11), in conjunction with the *fieldname_for_permitted_value* attribute.

NOTE A *Defined_Value_Domain* is like an *Enumerated_Value_Domain* (11.3.2.6), except the enumerated values are referenced from the external source that specifies the *Defined_Conceptual_Domain* (11.3.2.11), and the permissible values could or could not be imported into the registry. If the permissible values are imported, then they are imported as a set, so the begin_date and end_date attributes are specified on the *Defined_Value_Domain* instead of on the individual *Permissible_Values* (11.3.2.7).

Each *Defined_Value_Domain* class may participate in the association:

— *defined_permissible_value_set* (11.3.3.10) with zero or more *Defined_Permissible_Values* (11.3.2.14) which specify the values within the domain.

11.3.2.13.3 Attributes of Defined_Value_Domain

11.3.2.13.3.1 fieldname_for_permitted_value

Attribute name: *fieldname_for_permitted_value*

Definition: the name of the field at the external source where the permitted values may be accessed

Obligation: Mandatory

Multiplicity: *1*

Datatype: *String* (6.2.11)

11.3.2.13.3.2 begin_date

Attribute name: *begin_date*

Definition: date at which the *Defined_Value_Domain* became valid

Obligation: Mandatory

Multiplicity: *1*

Datatype: *Date* (6.2.3)

NOTE By imputation, this is also considered to be date at which the associated *Defined_Permissible_Value*s were bound to the associated *Defined_Value_Meaning*s, since the *defined_permissible_value_meaning* association mandates that there must be exactly one meaning (*Value_Meaning*) for each representation (*Permissible_Value*).

11.3.2.13.3.3 end_date

Attribute name: *end_date*

Definition: date at which the *Defined_Value_Domain* ceased to be valid

Obligation: Optional

Multiplicity: *0..1*

Datatype: *Date* (6.2.3)

NOTE 1 By imputation, this is also considered to be date at which the associated *Defined_Permissible_Value*s ceased to be bound to their associated meanings (*Defined_Value_Meaning*) via the *defined_permissible_value_meaning* association.

NOTE 2 The absence of the *permissible_value_end_date* attribute indicates that the *Permissible_Value* is still valid and (by imputation) still bound to its *Value_Meaning* via the *value_meaning* association.

—— End of attributes of *Defined_Value_Domain* ——

11.3.2.14 Defined_Permissible_Value class

11.3.2.14.1 Description of Defined_Permissible_Value

Defined_Permissible_Value is a class each instance of which models a *permissible value* (3.2.97), the *designation* (3.2.52) of a *value meaning* (3.2.142). A *Defined_Permissible_Value* is an expression of a *Defined_Value_Meaning* (11.3.2.12) within zero or more *Defined_Value_Domain*s (11.3.2.13).

Each *Defined_Permissible_Value* shall participate in the following association:

— defined_permissible_value_meaning (11.3.3.9) with exactly one Defined_Value_Meaning (11.3.2.12).

Each *Permissible_Value* may participate in the following association:

— *defined_permissible_value_set* (11.3.3.10) with zero or more *Defined_Value_Domains* (11.3.2.13). It is one of a set of such values that comprises an *Enumerated_Value_Domain* (11.3.2.6).

11.3.2.14.2 Attributes of Defined_Permissible_Value

11.3.2.14.2.1 permitted_value

Attribute name: *permitted_value*

Definition: the actual value of the *Permissible_Value*

Obligation: Mandatory

Multiplicity: *1*

Datatype: *Value* (6.2.13)

—— End of attributes of *Defined_Permissible_Value* ——

11.3.3.6

At the end of 11.3.3.6, insert subclauses 11.3.3.7 through 11.3.3.10:

11.3.3.7 domain_definition

The association domain_definition binds a set of *Conceptual_Domain_Definitions* (11.3.2.10) to a *Defined_Conceptual_Domain* (11.3.2.11). The association has two roles:

— *defined_domain* (verb form: defines) which references a *Defined_Conceptual_Domain;*

— *definition* (verb form: defined by) which references a *Conceptual_Domain_Definition*.

11.3.3.8 defined_value_meaning_set

The association *defined_value_meaning_set* binds a set of *Defined_Value_Meaning*s (11.3.2.12) to a *Defined_Conceptual_Domain* (11.3.2.11). The association has two roles:

— *containing_domain* (verb form: contained_in) which references a *Defined_Conceptual_Domain;*

— *member* (verb form: has_member) which references a *Defined_Value_Meaning*.

Each *member* (*Defined_Value_Meaning*) shall have one or more *containing_domains* (*Defined_Conceptual_Domain*). Each *containing_domain* (*Defined_Conceptual_Domain*) may have one or more *members* (*Defined_Value_Meanings*). The *defined_value_meaning_set* association is a weak containment association, which means that deletion of the containing *Enumerated_Conceptual_Domain* does not imply a cascading delete of the contained *Defined_Value_Meanings*, provided the *Defined_Value_Meaning* is shared with another *Defined_Conceptual_Domain*.

11.3.3.9 defined_permissible_value_meaning

The association *defined_permissible_value_meaning* binds one or more *Defined_Permissible_Values* (11.3.2.14) with a *Defined_Value_Meaning* (11.3.2.12). The association has two roles:

— *meaning* (verb form: has meaning) which references a *Defined_Value_Meaning;*

— *representation* (verb form: has representation) which references a *Defined_Permissible_Value*.

Each *representation* (*Defined_Permissible_Value*) shall have exactly one *meaning* (*Defined_Value_Meaning*). However, each *meaning* (*Defined_Value_Meaning*) may have zero or more *representations* (*Defined_Permissible_Values*).

11.3.3.10 defined_permissible_value_set

The association *defined_permissible_value_set* binds a set of *Defined_Permissible_Value*s (11.3.2.14) to a *Defined_Value_Domain* (11.3.2.13). The association has two roles:

— *member* (verb form: has member) which references a *Defined_Permissible_Value*;

— *containing_domain* (verb form: contains_domain) which references a *Defined_Value_Domain*.

Each *member* (*Defined_Permissible_Value*) may have zero or more *containing_domains* (*Defined_Value_Domains*). Each *containing_domain* (*Defined_Value_Domain*) may have one or more *members* (*Defined_Permissible_Values*). The *permissible_value_set* association is a weak containment relation, i.e., deletion of the containing domain does not cause a cascading delete of the members (*Defined_Permissible_Values*). Thus it is possible to associate *defined permissible values* with *defined value meanings* without defining a complete *defined value domain*.

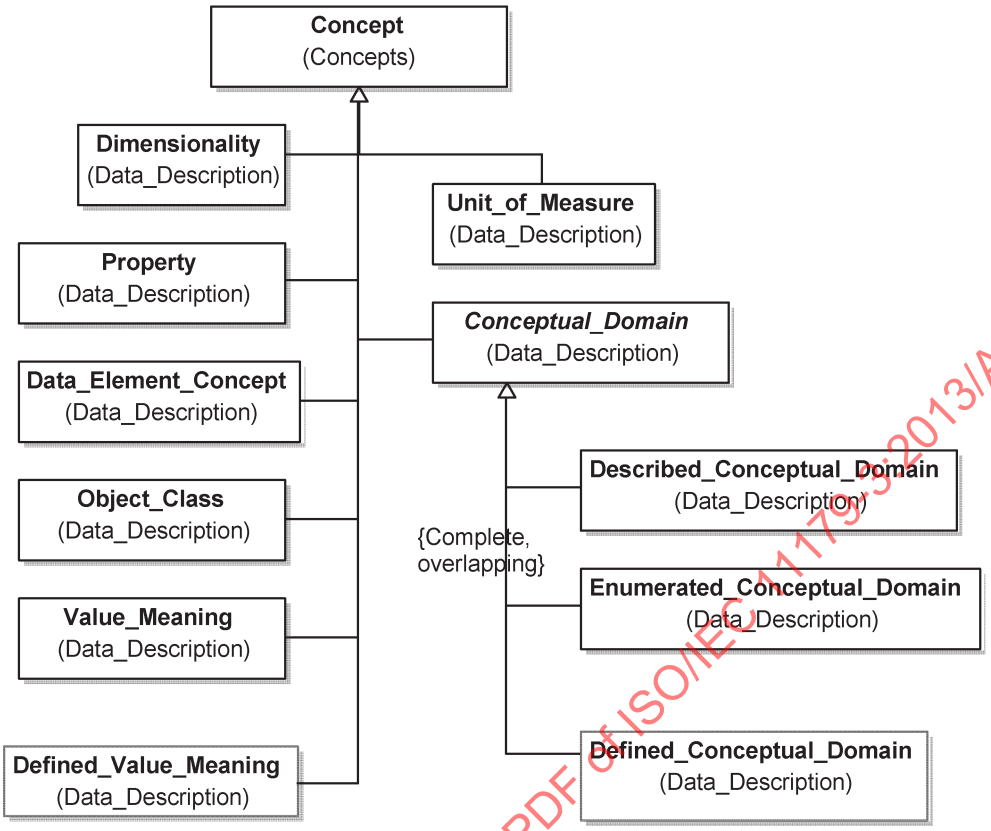
11.5.1

Insert the following NOTE after Figure 15.

NOTE ISO/IEC 11179-3:2003 included an explicit depiction of Representation_Class in the Data_Element metamodel region. In ISO/IEC 11179-3:2013, this explicit representation has been dropped in favour of treating Representation_Class as a type of Classification_Scheme, which in turn is now considered a type of Concept_System. The Classification metamodel region is described in 9.2. Annex F describes modelling Representation_Class as a Concept_System.

11.7, Figure 17

Replace Figure 17 with the following:



11.7

At the end of 11.7, insert 11.8:

11.8 Value_Domain_Subset metamodel region

11.8.1 Overview

This metamodel region (Figure 18) shows how subsets of a *Value_Domain* may be used for *Data_Elements* where some of the values in the full Value_Domain are not applicable.

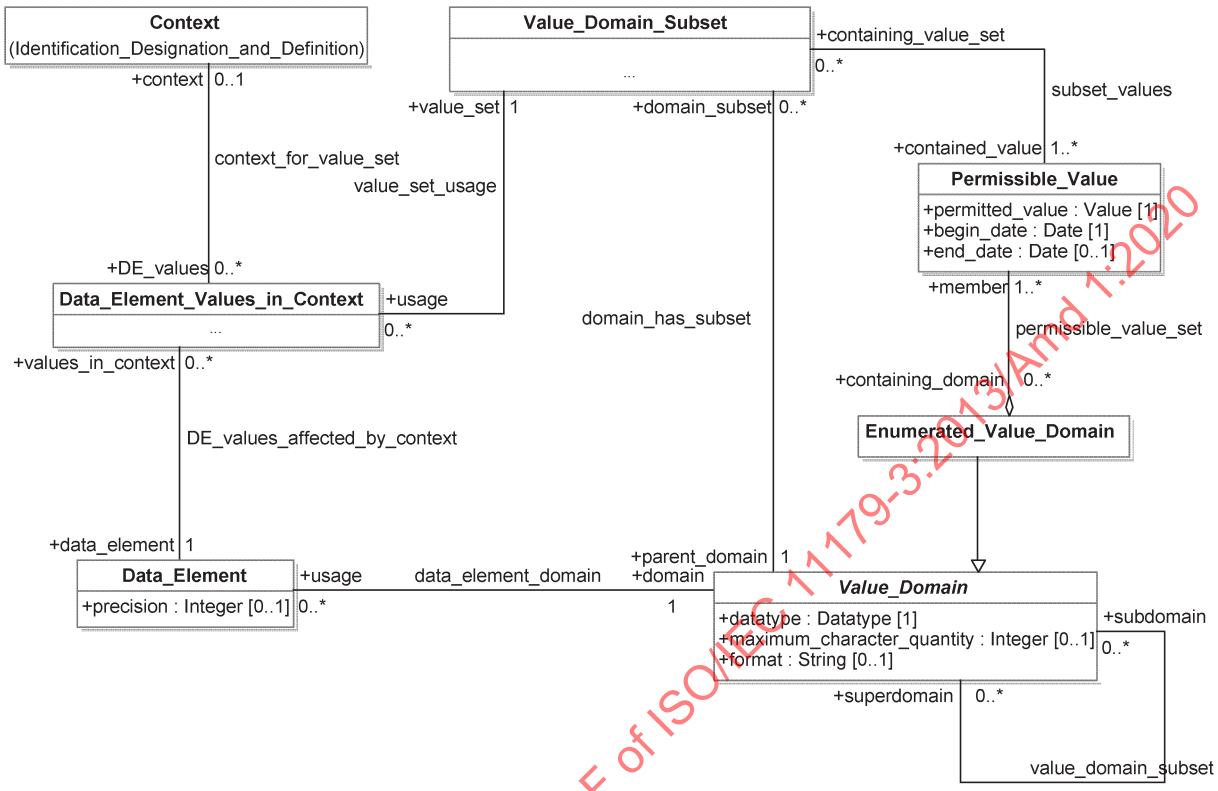


Figure 18 — Value_Domain_Subset metamodel region

11.8.2 Classes in the Value_Domain_Subset region

11.8.2.1 Context class

Context is described in 6.4.2.3 and 7.3.2.5. This region specifies an additional association.

Context may participate in the *context_for_value_set* association (11.8.3.1) with zero or more *Data_Element_Values_in_Context* (11.8.2.3).

11.8.2.2 Data_Element class

Data_Element is described in 11.5.2.1. This region specifies an additional association.

Data_Element may participate in a *DE_values_affected_by_context* association (11.8.3.2) with zero or more *Data_Element_Values_in_Contexts* (11.8.2.3).

11.8.2.3 Data_Element_Values_in_Context class

11.8.2.3.1 Description of Data_Element_Values_in_Context

Data_Element_Values_in_Context is a class each instance of which models the use of a *Value_Domain_Subset* (11.8.2.5) to represent the *Permissible_Values* (11.8.2.7) for a *Data_Element* (11.8.2.2) in zero or one *Contexts* (11.8.2.1).

11.8.2.4 Value_Domain class

Value_Domain is described in 11.3.2.5.

11.8.2.5 Value_Domain_Subset class

11.8.2.5.1 Description of Value_Domain_Subset

Value_Domain_Subset is a class each instance of which models a subset of the *Permissible_Values* (11.8.2.7) in a *Value_Domain* (11.8.2.4).

11.8.2.6 Enumerated_Value_Domain class

Enumerated_Value_Domain is described in 11.3.2.6.

11.8.2.7 Permissible_Value class

Permissible_Value is described in 11.3.2.7. This region specifies an additional association.

Permissible_Value may participate in a *subset_values* association (11.8.3.6) with zero or more *Value_Domain_Subsets* (11.8.2.5).

11.8.3 Associations in the Value_Domain_Subset region

11.8.3.1 context_for_value_set

The association *context_for_value_set* records the use of a *Context* (11.8.2.1) by zero or more *Data_Element_Values_in_Context* (11.8.2.3). The association has two roles:

— *context* (verb form: in context) which references the *Context*;

— *DE_values* (verb form: constrains DE values) which references the *Data_Element_Values_in_Context* that associates the applicable *Value_Domain_Subset* (11.8.2.5) and *Data_Element* (11.8.2.2).

11.8.3.2 DE_values_affected_by_context

The association *DE_values_affected_by_context* records the use of zero or more *Data_Element_Values_in_Context* (11.8.2.3) by a *Data_Element* (11.8.2.2). The association has two roles:

— *data_element* (verb form: is data element) which references the *Data_Element*;

— *values_in_context* (verb form: has values) which references the *Data_Element_Values_in_Context* that associates the applicable *Value_Domain_Subset* (11.8.2.5) and *Context* (11.8.2.1).

11.8.3.3 data_element_domain

data_element_domain is described in 11.1.3.2.

11.8.3.4 domain_has_subset

The association *domain_has_subset* records a superset-subset relationship between a *Value_Domain* (11.8.2.4) and a *Value_Domain_Subset* (11.8.2.5). The association has two roles:

— *parent_domain* (verb form: is parent domain) which references the *Value_Domain*;

— *domain_subset* (verb form: is domain subset) which references the *Value_Domain_Subset*.

11.8.3.5 permissible_value_set

permissible_value_set is described in 11.3.3.5.

11.8.3.6 subset_values

The association *subset_values* binds a *Value_Domain_Subset* (11.8.2.5) with a set *Permissible_Values* (11.8.2.7). The association has two roles:

— *containing_value_set* (verb form: is containing value set) which references the *Value_Domain_Subset*;

— *contained_value* (verb form: contains value) which references the *Permissible_Value*.

