



BSI Standards Publication

Information technology — Metadata registries (MDR)

Part 31: Metamodel for data specification registration

National foreword

This British Standard is the UK implementation of ISO/IEC 11179-31:2023. Together with BS ISO/IEC 11179-3:2023, it supersedes BS ISO/IEC 11179-3:2013+A1:2020, which will be withdrawn on publication of BS ISO/IEC 11179-3:2023.

The UK participation in its preparation was entrusted to Technical Committee IST/40, Data management and interchange.

A list of organizations represented on this committee can be obtained on request to its committee manager.

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2023
Published by BSI Standards Limited 2023

ISBN 978 0 539 14622 6

ICS 35.040.50

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 January 2023.

Amendments/corrigenda issued since publication

Date	Text affected
<hr/>	

**Information technology — Metadata
registries (MDR) —**

**Part 31:
Metamodel for data specification
registration**

Technologies de l'information — Registres de métadonnées (RM) —

*Partie 31: Métamodèle pour l'enregistrement des spécifications de
données*



Reference number
ISO/IEC 11179-31:2023(E)

AVA Documents
Document Sharing Hub
© ISO/IEC 2023



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2023

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

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

Published in Switzerland

Contents

Page

Foreword	vii
Introduction	viii
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Abbreviated terms	6
5 Conformance	6
5.1 Overview of conformance	6
5.2 Degree of conformance	6
5.2.1 General	6
5.2.2 Strictly conforming implementations	7
5.2.3 Conforming implementations	7
5.3 Conformance by feature	7
5.4 Registry conformance	8
5.4.1 Standard profiles for edition 4 registries	8
5.4.2 Conformance labels	8
5.5 Implementation conformance statement (ICS)	8
5.6 Obligation	8
6 Relationship to ISO/IEC 11179-3	9
6.1 Metamodel for a metadata registry	9
6.2 Specification of the metamodel	9
6.3 Use of UML Class diagrams and textual description	9
6.4 Package dependencies	10
6.5 Subclassing the Constraint_Set class	10
7 Data_Specification package	11
7.1 Overview of the Data_Specification package	11
7.2 High-level Data Specification metamodel region	12
7.2.1 Overview of the high-level metamodel region	12
7.2.2 Classes in the High-level Data Specification metamodel region	13
7.2.3 Associations of the High-Level Data Specification metamodel region	16
7.2.4 Constraints of the High Level metamodel region	17
7.3 Data Element Concept metamodel region	18
7.3.1 Overview of the Data Element Concept metamodel region	18
7.3.2 Classes in the Data Element Concept metamodel region	19
7.3.3 Associations in the Data Element Concept metamodel region	22
7.4 Conceptual Domain and Value Domain metamodel region	22
7.4.1 Overview of the Conceptual Domain and Value Domain metamodel region	22
7.4.2 Classes in the Conceptual Domain and Value Domain metamodel region	24
7.4.3 Association Classes in the Conceptual Domain and Value Domain metamodel region	41
7.4.4 Associations in the Conceptual Domain and Value Domain metamodel region	44
7.4.5 Additional Constraints of the Conceptual Domain and Value Domain metamodel region	45
7.5 Measurement metamodel region	46
7.5.1 Overview of the Measurement metamodel region	46
7.5.2 Classes in the Measurement metamodel region	47
7.5.3 Associations in the Measurement metamodel region	50
7.6 Data Element metamodel region	51
7.6.1 Overview of the Data Element metamodel region	51
7.6.2 Classes in the Data Element metamodel region	52
7.6.3 Associations in the Data Element metamodel region	52

7.7	Value Domain Subset metamodel region.....	57
7.7.1	Overview of the Value Domain Subset metamodel region.....	57
7.7.2	Classes in the Value Domain Subset metamodel region.....	57
7.7.3	Association classes in the Value Domain Subset metamodel region.....	61
7.7.4	Associations in the Value Domain Subset metamodel region.....	62
7.7.5	Constraints in the Value Domain Subset metamodel region.....	63
7.8	Conceptual Domain Subset metamodel region.....	63
7.8.1	Overview of the Conceptual Domain Subset metamodel region.....	63
7.8.2	Classes in the Conceptual Domain Subset metamodel region.....	64
7.8.3	Association classes in the Conceptual Domain Subset metamodel region.....	66
7.8.4	Associations in the Conceptual Domain Subset metamodel region.....	67
7.8.5	Constraints in the Conceptual Domain Subset metamodel region.....	68
7.9	Composite Data Elements and Datatypes metamodel region.....	68
7.9.1	Overview of the Composite Data Elements and Datatypes metamodel region.....	68
7.9.2	Classes in the Composite Data Elements and Datatypes metamodel region.....	69
7.9.3	Association Class in the Composite Data Elements and Datatypes metamodel region.....	71
7.9.4	Associations in the Composite Data Elements and Datatypes metamodel region.....	72
Annex A (informative) Consolidated class hierarchy		73
Annex B (informative) Example of Address as Composite Data Element and Datatype		74
Annex C (informative) Partial consolidated data specification metamodel.....		75
Bibliography		76

List of Figures

Figure 1 — Package dependencies.....	10
Figure 2 — Subclasses of Constraint_Set.....	11
Figure 3 — High-level Data Specification metamodel.....	12
Figure 4 — Data Element Concept metamodel region.....	18
Figure 5 — Conceptual Domain and Value Domain metamodel region	23
Figure 6 — Measurement metamodel region.....	46
Figure 7 — Data Element metamodel region	51
Figure 8 — Value Domain Subset metamodel region.....	57
Figure 9 — Conceptual Domain Subset metamodel region.....	63
Figure 10 — Composite Data Elements and Datatypes metamodel region.....	69
Figure A.1 — Consolidated Class Hierarchy.....	73
Figure B.1 — Object diagram of Address as Composite Datatype and Data Element.....	74
Figure C.1 — Partial Consolidated Data Specification metamodel	75

List of Tables

Table 1 — Attributes of Data_Element_Constraint_Set class	21
Table 2 — Attributes of Conceptual_Domain class	24
Table 3 — Attributes of Enumerated_Conceptual_Domain class	26
Table 4 — Attributes of Enumerated_Conceptual_Domain_Definition class	29
Table 5 — Attributes of Value_Domain class	30
Table 6 — Attributes of Permissible_Value class	33
Table 7 — Attributes of Reference_Enumerated_Value_Domain class	34
Table 8 — Attributes of Datatype class	35
Table 9 — Example 1 of Datatype class	35
Table 10 — Example 2 of Datatype class	35
Table 11 — Attributes of Datatype_Scheme class	36
Table 12 — Attributes of Reference_Enumerated_CD_Constraint_Set class	37
Table 13 — Attributes of Local_Enumerated_CD_Constraint_Set class	38
Table 14 — Attributes of Value_Domain_Constraint_Set class	38
Table 15 — Attributes of Described_Value_Domain_Constraint_Set class	39
Table 16 — Attributes of Reference_Enumerated_VD_Constraint_Set class	40
Table 17 — Attributes of Local_Enumerated_VD_Constraint_Set class	40
Table 18 — Attributes of Permissible_Value_Constraint_Set class	41
Table 19 — Attributes of Value_Meaning_Set association class	42
Table 20 — Attributes of Permissible_Value_Meaning association class	43
Table 21 — Attributes of Permissible_Value_Set association class	44
Table 22 — Attributes of Dimensionality class	50
Table 23 — Attributes of Data_Element class	52
Table 24 — Attributes of Data_Element_Example class	53
Table 25 — Attributes of Data_Element_Derivation_Rule class	54
Table 26 — Attributes of Data_Element_Constraint_Set class	55
Table 27 — Attributes of Local_Enumerated_Value_Domain_Subset class	60
Table 28 — Attributes of Local_Enumerated_VD_Subset_Constraint_Set class	61
Table 29 — Attributes of Subset_Values association class	62
Table 30 — Attributes of Local_Enumerated_Conceptual_Domain_Subset class	65
Table 31 — Attributes of Local_Enumerated_CD_Subset_Constraint_Set class	66
Table 32 — Attributes of Subset_Value_Meanings association class	67
Table 33 — Attributes of Data_Element_Collection class	70
Table 34 — Attributes of Data_Element_Sequence association class	71

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 or www.iec.ch/members_experts/refdocs).

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 <https://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. In the IEC, see www.iec.ch/understanding-standards.

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 and IEC websites.

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 and www.iec.ch/national-committees.

Introduction

In ISO/IEC 11179-3, the structure of a Metadata Registry is specified in the form of a conceptual data model. ISO/IEC 11179-3 specifies a metamodel for “registry common facilities”, which is intended to be extended by other parts of ISO/IEC 11179 for specific purposes.

This document provides a specification of the extensions to the registry metamodel specified in ISO/IEC 11179-3 to enable the registration of metadata about data elements and associated concepts, such as “data element concepts”, “conceptual domains” and “value domains”. Generically, these are all referred to as “metadata items”. Such metadata are necessary to clearly describe, record, analyse, classify and administer data.

This document is part of the 4th edition modularization of the ISO/IEC 11179 series. It extracts the Data Description package from ISO/IEC 11179-3:2013 to make it more accessible and renames it “Metamodel for data specification registration”. At the same time, some enhancements have been made as follows:

- support for externally defined “reference enumerated conceptual domains” ([7.4.2.6](#)) and “reference enumerated value domains” ([7.4.2.13](#));
- support for sub-setting of value domains ([7.7](#)) and conceptual domains ([7.8](#)) within a specified context;
- support for composite data elements and data types ([7.9](#));
- finer-grained conformance options (see [5.3](#));
- relaxation of some constraints in the standard, while giving registration authorities the ability to enforce them if they wish (see [6.5](#)).

From [Clause 5](#) onwards, this document uses:

- **bold** font to highlight terms which represent metadata objects specified by the metamodel;
- normal font for terms which represent concepts defined in [Clause 3](#).

EXAMPLE **Conceptual_Domain** ([7.2.2.2](#)) is a class each instance of which models a conceptual domain.

Information technology — Metadata registries (MDR) —

Part 31:

Metamodel for data specification registration

1 Scope

This document provides a specification for an extension to a Metadata Registry (MDR), as specified in ISO/IEC 11179-3, in which metadata that describes data elements and associated concepts, such as “data element concepts”, “conceptual domains” and “value domains” can be registered.

The specification in this document, together with the relevant clauses of the specification in ISO/IEC 11179-3, provides the ability to record metadata about:

- a) data elements, units of measure and derivation rules;
- b) data element concepts and associated object classes and properties;
- c) conceptual domains, conceptual domain subsets and value meanings;
- d) value domains, value domain subsets, datatypes and permissible values.

This document is applicable to the formulation of data representations, concepts, meanings and relationships to be shared among people and machines, independent of the organization that produces the data. It is not applicable to the physical representation of data as bits and bytes at the machine level.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 11179-3:2023, *Information technology — Metadata registries (MDR) — Part 3: Metamodel for registry common facilities*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 11179-3 and the following apply.

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

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>

3.1

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

3.2

property

quality common to all members of an *object class* (3.1)

3.3

characteristic

abstraction of a *property* ([3.2](#))

EXAMPLE 'Having a cable for connecting with a computer' as a characteristic of the concept 'cord mouse'.

Note 1 to entry: Characteristics are used for describing *concepts* ([3.4](#)).

[SOURCE: ISO 1087:2019, 3.2.1]

3.4

concept

unit of knowledge created by a unique combination of *characteristics* ([3.3](#))

Note 1 to entry: Concepts are not necessarily bound to particular natural languages. They are, however, influenced by the social or cultural background which often leads to different categorizations.

Note 2 to entry: A concept is independent of its representation.

[SOURCE: ISO 1087:2019, 3.2.7, modified — Note 2 to entry changed.]

3.5

conceptual domain

CD

concept ([3.4](#)) whose meaning is expressed as an enumerated set, a description of subordinate concepts or both, which are *value meanings* ([3.10](#))

3.6

described conceptual domain

conceptual domain ([3.5](#)) that is specified by a description or specification, such as a rule, a procedure or a range (i.e. interval)

3.7

enumerated conceptual domain

conceptual domain ([3.5](#)) that is specified by a list of all its *value meanings* ([3.10](#))

Note 1 to entry: No ordering of the value meanings is implied.

3.8

local enumerated conceptual domain

enumerated conceptual domain ([3.7](#)) whose *value meanings* ([3.10](#)) are enumerated locally within the registry

Note 1 to entry: c.f. *reference enumerated conceptual domain* ([3.11](#)).

3.9

local enumerated conceptual domain subset

subset of the *value meanings* ([3.10](#)) in a *local enumerated conceptual domain* ([3.8](#)) used to restrict the value meanings a *data element concept* ([3.25](#)) can assume in a particular context

3.10

value meaning

semantic content of a value

Note 1 to entry: The representation of value meanings in a *registry* ([3.36](#)) shall be independent of (and shall not constrain) their representation in any corresponding *value domain* ([3.13](#)).

3.11

reference enumerated conceptual domain

enumerated conceptual domain ([3.7](#)) that is specified by a formal definition

Note 1 to entry: The definition may reference externally enumerated *value meanings* ([3.10](#)).

3.12**enumerated conceptual domain definition**

formal definition of an *enumerated conceptual domain* (3.7)

Note 1 to entry: The definition may reference externally enumerated *value meanings* (3.10).

3.13**value domain****VD**

set of *permissible values* (3.19)

Note 1 to entry: The *value domain* provides representation but has no implication as to what *data element concept* (3.25) the values are associated with nor what the values mean.

Note 2 to entry: The *permissible values* can either be enumerated, expressed via a description, or a combination of the two.

3.14**described value domain**

value domain (3.13) that is specified by a description or specification, such as a rule, a procedure or a range (i.e. interval)

3.15**enumerated value domain**

value domain (3.13) that is specified by a list of all its *permissible values* (3.19)

Note 1 to entry: No ordering of the permissible values is implied.

3.16**local enumerated value domain**

enumerated value domain (3.15) whose *permissible values* (3.19) are stored within the registry

Note 1 to entry: c.f. *reference enumerated value domain* (3.18).

3.17**local enumerated value domain subset**

subset of the *permissible values* (3.19) in a *local enumerated value domain* (3.16) used to restrict the value meanings a *data element* (3.23) can assume in a particular context

3.18**reference enumerated value domain**

enumerated value domain (3.15) that is specified by reference to an external specification, including externally enumerated *permissible values* (3.19)

3.19**permissible value**

designation of a *value meaning* (3.10)

Note 1 to entry: Permissible values may be specified either as part of a *value domain* (3.13) or only associated with a *value meaning* (3.10).

Note 2 to entry: Within a value domain, permissible values can either be enumerated, expressed via a description, or a combination of the two.

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, it is possible for the described meaning to be associated with a range of values, e.g. weight in kilograms.

3.20

data

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

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

[SOURCE: ISO/IEC 2382:2015, 2121272, modified — Notes to entry 2 and 3 deleted.]

3.21

datatype

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

[SOURCE: ISO/IEC 11404:2007, 3.12]

3.22

datatype scheme

source of the specification of one or more *datatypes* ([3.21](#))

[SOURCE: ISO/IEC 11404:2007, 3.12]

3.23

data element

⟨organization of data⟩ unit of *data* ([3.20](#)) that is considered in context to be indivisible

Note 1 to entry: The definition states that a data element is “indivisible” in some context. This means that it is possible that a data element considered indivisible in one context (e.g. telephone number) may be divisible in another context, (e.g. country code, area code, local number).

EXAMPLE The data element “age of a person” with values consisting of all combinations of 3 decimal digits.

[SOURCE: ISO/IEC 2382:2015, 2121599, modified — Example moved to the end without the Note to entry prefix. Other Notes to entry have been replaced.]

3.24

data element collection

one or more *data elements* ([3.23](#)) that may be unordered or ordered

Note 1 to entry: Examples of unordered collections are a set or a bag (or multiset). An example of an ordered collection is a list.

3.25

data element concept

concept ([3.4](#)) that can be represented in the form of a *data element* ([3.23](#)), described independently of any particular representation

Note 1 to entry: A data element concept is implicitly associated with both the property and the object class whose combination it expresses.

Note 2 to entry: A data element concept may also be associated with zero, one or more *conceptual domains* ([3.5](#)) each of which expresses its *value meanings* ([3.10](#)).

Note 3 to entry: A data element concept may also be associated with zero, one or more *data elements* ([3.23](#)) each of which provide representation for the data element concept via its associated *value domain* ([3.13](#)).

3.26

data element derivation

application of a *data element derivation rule* ([3.28](#)) to one or more input *data elements* ([3.23](#)) to derive one or more output data elements

3.27

data element example

representative illustration of a *data element* ([3.23](#))

3.28**data element derivation rule**

logical, mathematical, other operations or some combination specifying derivation

3.29**dimensionality**

set of equivalent *units of measure* (3.33)

Note 1 to entry: 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.

Note 2 to entry: 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:

- a) given a value measured in degrees Fahrenheit there is a value measured in degrees Celsius with the same quantity, and vice-versa, by the well-known correspondences $^{\circ}\text{C} = (5/9) * (^{\circ}\text{F} - 32)$ and $^{\circ}\text{F} = (9/5) * (^{\circ}\text{C}) + 32$.
- b) the same operations can be performed on both values.

Note 3 to entry: The units of measure "temperature in degrees Celsius" and "temperature in kelvins" do not belong to the same dimensionality. Even though it is easy to convert quantities from one unit of measure to the other ($^{\circ}\text{C} = \text{K} - 273,15$ and $\text{K} = ^{\circ}\text{C} + 273,15$), the characterizing operations in kelvins include taking ratios, whereas this is not the case for degrees Celsius. For instance, 20 K is twice as warm as 10 K, but 20 $^{\circ}\text{C}$ is not twice as warm as 10 $^{\circ}\text{C}$.

Note 4 to entry: 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.

Note 5 to entry: Quantities may be grouped together into categories of quantities which are mutually comparable. Lengths, diameters, distances, heights, wavelengths and so on would constitute such a category. Mutually comparable quantities have the same dimensionality. ISO 80000-1 [13] calls these "quantities of the same kind".

Note 6 to entry: 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). The present concept of dimensionality equates to what ISO 80000-1 calls Dimensional Product, rather than to Dimension.

3.30**measure class**

set of equivalent *units of measure* (3.33) for association with one or more *dimensionalities* (3.29)

3.31**coordinate**

measurement from the origin of a frame of reference

3.32**notation**

formal syntax and associated semantics

EXAMPLE UML, MOF, OCL, OWL/RDF, SKOS, CGIF, XCL, XTM or ISO/IEC 11404

Note 1 to entry: Formal syntax is often intended for machine processing.

[SOURCE: ISO/IEC 11179-3:2023, 3.2.36]

3.33**unit of measure**

$\langle$ value domain $\rangle$ actual units in which the associated values are measured

Note 1 to entry: ISO 80000-1 [13] specifies a system of physical measurement (the International System of Units, SI). Physical measurement is only one type of measurement. Value measurement is another type of measurement. This document permits the use of any appropriate system of measurement.

Note 2 to entry: The *dimensionality* (3.29) of the associated *conceptual domain* (3.5) shall be appropriate for the specified *unit of measure*.

3.34

unit of measure dimensionality

dimensionality (3.29) that specifies the equivalence relation that applies to all values representing a particular unit

3.35

quantity

value associated with a *unit of measure* (3.33)

Note 1 to entry: 32° Fahrenheit and 0° Celsius are quantities, and they are equivalent values in different measuring systems.

3.36

registry

information system for *registration* (3.37)

[SOURCE: ISO/IEC 11179-1:2023, 3.2.34]

3.37

registration

set of rules, operations and procedures for inclusion of an item in a *registry* (3.36)

Note 1 to entry: A detailed description of registration as it applies in ISO/IEC 11179 is found in ISO/IEC 11179-6.

[SOURCE: ISO/IEC 11179-1:2023, 3.2.88]

4 Abbreviated terms

CD	conceptual domain
DE	data element
DEC	data element concept
UML	Unified Modeling Language
URI	Universal Resource Identifier
VD	value domain
XML	eXtensible Markup Language

5 Conformance

5.1 Overview of conformance

Conformance rules for a metadata registry are specified in ISO/IEC 11179-3:2023, Clause 4. The subclause “Degree of conformance” is repeated here for convenience. The subsequent subclauses extend the rules from ISO/IEC 11179-3.

5.2 Degree of conformance

5.2.1 General

The distinction between “strictly conforming” and “conforming” implementations is necessary to address the simultaneous needs for interoperability and extensions. This document describes

specifications that promote interoperability. Extensions are motivated by needs of users, vendors, institutions and industries, and:

- a) are not directly specified by this document;
- b) are specified and agreed to outside this document;
- c) may serve as trial usage for future editions of this document.

A strictly conforming implementation can be limited in usefulness but is maximally interoperable with respect to this document. A conforming implementation can be more useful but can be less interoperable with respect to this document.

5.2.2 Strictly conforming implementations

A strictly conforming implementation:

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

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

5.2.3 Conforming implementations

A conforming implementation:

- a) shall support all mandatory, optional and conditional classes, attributes, datatypes and associations;
- b) as permitted by the implementation, may use, test, access or probe for extension features or extensions to classes, attributes, datatypes, associations or any combination thereof;
- c) may recognize, act on or allow the production of classes, attributes, datatypes, associations or any combination thereof that are dependent on implementation-defined behaviour.

NOTE 1 All strictly conforming implementations are also conforming implementations.

NOTE 2 The use of extensions to the metamodel can cause undefined behaviour.

5.3 Conformance by feature

Conformance claims may be made to the whole of [Clause 7](#) or to specific features within that clause. [Clause 7](#) is dependent upon one or more clauses of ISO/IEC 11179-3, so conformance to all or part of [Clause 7](#) shall be understood to imply conformance also to relevant provisions specified in one or more of the clauses in ISO/IEC 11179-3.

A conformance statement shall specify exactly the features supported and not supported.

5.4 Registry conformance

5.4.1 Standard profiles for edition 4 registries

This document specifies the following standard profiles in addition to those specified in ISO/IEC 11179-3:2023, 4.4.2.

- **Data Specification Registry:** Implements [Clause 7](#) of this document, in addition to all provisions of the “Basic Registry” profile of ISO/IEC 11179-3:2023, 4.4.2.
- **Data Specification Registry with mapping:** Implements [Clause 7](#) of this document, in addition to all provisions of the “Basic Registry with mapping” profile of ISO/IEC 11179-3:2023, 4.4.2.
- **Conceptual and Value Domain Registry:** Implements [7.3](#) of this document, in addition to all the provisions of the “Basic Registry” profile of ISO/IEC 11179-3:2023, 4.4.2.
- **Conceptual and Value Domain Registry with mapping:** Implements [7.3](#) of this document, in addition to all the provisions of the “Basic Registry with mapping” profile of ISO/IEC 11179-3:2023, 4.4.2.

5.4.2 Conformance labels

Conformance to the profiles specified in [5.4.1](#) may be claimed using the following labels, respectively:

- ISO/IEC 11179-31:2023 Data Specification Registry
- ISO/IEC 11179-31:2023 Data Specification Registry with mapping
- ISO/IEC 11179-31:2023 Conceptual and Value Domain Registry
- ISO/IEC 11179-31:2023 Conceptual and Value Domain Registry with mapping.

5.5 Implementation conformance statement (ICS)

An implementation claiming conformance to this document shall include an Implementation Conformance Statement stating:

- whether it conforms or strictly conforms;
- which clauses are or are not supported;
- what extensions, if any, are supported or used.

A standard profile may be referenced, if applicable.

EXAMPLE Product Y conforms to ISO/IEC 11179-31:2023 Data Specification Registry with Mapping, except that it relies on the Item Mapping facility of ISO/IEC 11179-3:2023 in place of implementing the ISO/IEC 11179-31:2023 Data Element Derivation feature ([7.6.2.5](#), [7.6.2.6](#), [7.6.3.4](#), [7.6.3.5](#), [7.6.3.6](#)).

5.6 Obligation

Properties and relationships specified in this document are one of: “mandatory”, “conditional” or “optional”. The obligation is not explicitly stated but is to be inferred from the multiplicity of the property or relationship, and the presence or absence of a condition. In addition, a registration authority can specify additional constraints to be applied to particular **Administered_Items** (ISO/IEC 11179-3:2023, 9.4.2), using **Constraint_Sets** (ISO/IEC 11179-3:2023, 9.4.5). (See [6.5](#).)

For the purpose of conformance:

- a) mandatory properties and relationships shall exist and shall conform to the provisions of this document;

- b) anything specified as Conditional within this document shall be treated as Mandatory if the associated condition is satisfied and shall otherwise be not present;
- c) optional properties and relationships are not required to exist, but if they do exist, they shall conform to the provisions of this document.

Such obligation is enforced if and only if the **Registration_Status** of the associated registry items is “Recorded” or higher (see ISO/IEC 11179-3:2023, 3.2.70 and 9.4.4.3; and ISO/IEC 11179-6:2023, 4.3.2).

6 Relationship to ISO/IEC 11179-3

6.1 Metamodel for a metadata registry

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

ISO/IEC 11179-3 uses a metamodel to describe the information model of a metadata registry. The registry in turn will be used to describe and model other data, for example about enterprise, public administration or business applications. The registry metamodel is specified as a conceptual data model, i.e. one that describes how relevant information is structured in the natural world. In other words, it is how the human mind is accustomed to thinking of the information.

6.2 Specification of the metamodel

The conventions used in specifying the metamodel are described in ISO/IEC 11179-3:2023, 5.3. Many of the classes specified in the Data Specification package (see [Clause 7](#)) inherit from **Item**, which is specified in the Core model region in ISO/IEC 11179-3:2023, 6.4.2.1. As **Items**, instances of these classes may be identified, registered, administered, named, defined and classified.

6.3 Use of UML Class diagrams and textual description

This document uses both text and UML class diagrams (References [\[10\]](#), [\[11\]](#)) to describe the metamodel. Both are normative and are intended to be complementary. However, if a conflict exists between what is specified in UML and what is specified in text, the text takes precedence until a correction is made to make them consistent. Further, if a conflict exists between a formal definition and other normative text, the formal definition takes precedence until a correction is made to make them consistent.

A consolidated UML class hierarchy is included as [Annex A](#).

While the model diagrams are presented in UML notation, this document does not assume nor endorse any specific system environment, database management system, database design paradigm, system development methodology, data definition language, command language, system interface, user interface, computing platform or any technology required for implementation.

6.4 Package dependencies

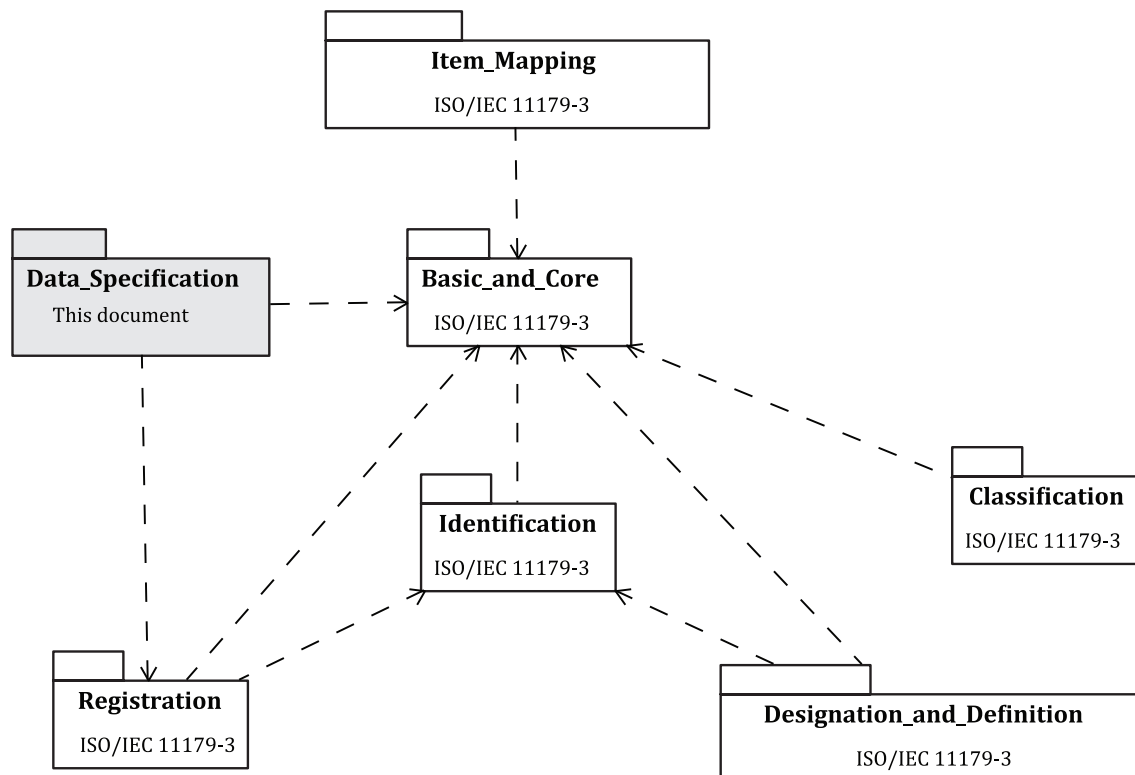


Figure 1 — Package dependencies

Figure 1 illustrates the dependencies among the packages. The lines in Figure 1 illustrate dependencies in the direction of the arrow. The Data Specification package is specified in Clause 7. The other packages shown in the figure are specified in ISO/IEC 11179-3. In order to implement a package that has dependencies, the packages on which it is dependent shall also be implemented. The dependencies are of three types:

- Subclassing from classes in another package, e.g. **Conceptual_Domain** (7.3.2.4), **Data_Element_Concept** (7.3.2.3), **Object_Class** (7.3.2.1) and **Property** (7.3.2.2) in the Data Element Concept region (7.3) of the Data_Specification package (Clause 7) are all subclassed from the **Concept** class in the Basic_and_Core package specified in ISO/IEC 11179-3:2023, Clause 6.
- Association between classes, e.g. **Registered_Item** in the Registration package in ISO/IEC 11179-3:2023, 9.4.1 has an association with **Reference_Document** in the Basic_and_Core package specified in ISO/IEC 11179-3:2023, Clause 6.
- Some attributes use a predefined datatype or a class from another package as a datatype e.g. the **contact** attribute of the **Stewardship_Record** class in the Registration package in ISO/IEC 11179-3:2023, 9.4.8 uses the **Contact** class of the Basic_and_Core package specified in ISO/IEC 11179-3:2023, Clause 6 as a datatype.

Conformance options are specified in Clause 5 and standard conformance profiles in 5.4.1.

6.5 Subclassing the Constraint_Set class

This document extends the **Constraint_Set** class (ISO/IEC 11179-3:2023, 9.4.5) by specifying subclasses to support constraints specified in this document. See Figure 2.

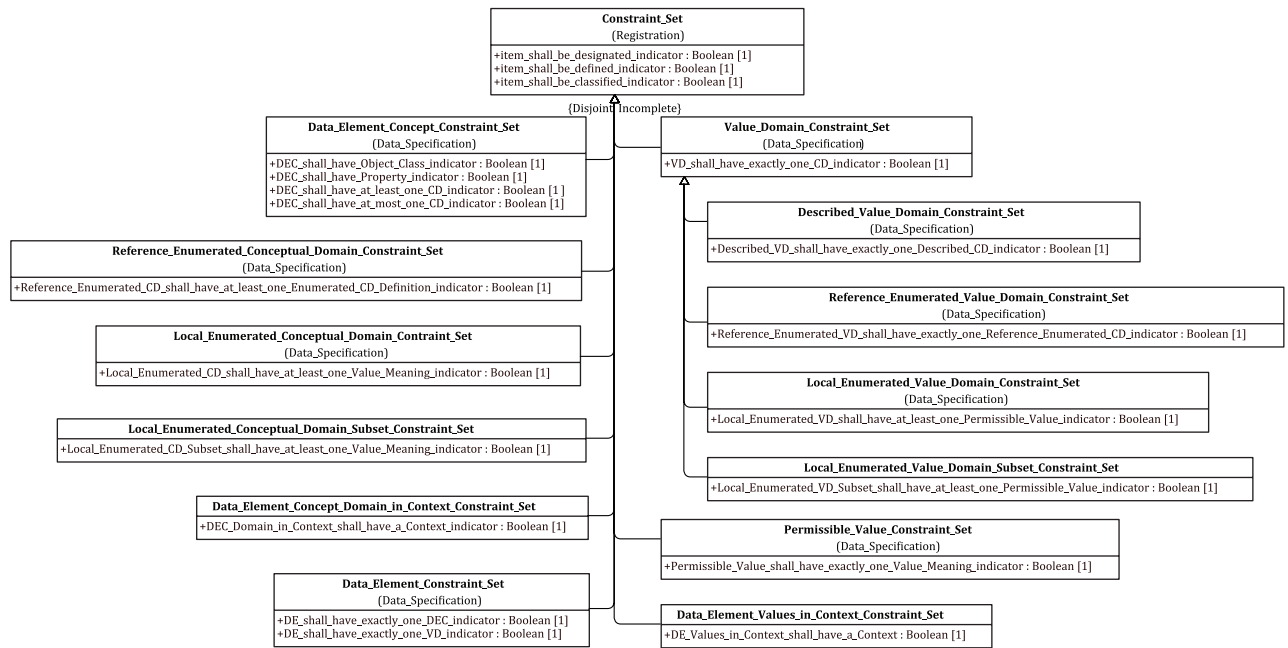


Figure 2 — Subclasses of Constraint_Set

For details, see the following subclasses:

- **Data_Element_Concept_Constraint_Set** class ([7.3.2.5](#));
- **Reference_Enumerated_CD_Constraint_Set** class ([7.4.2.16](#));
- **Local_Enumerated_CD_Constraint_Set** class ([7.4.2.17](#));
- **Local_Enumerated_CD_Subset_Constraint_Set** class ([7.8.2.8](#));
- **Data_Element_Constraint_Set** class ([7.6.2.7](#));
- **Value_Domain_Constraint_Set** class ([7.4.2.18](#));
- **Described_Value_Domain_Constraint_Set** class ([7.4.2.19](#));
- **Reference_Enumerated_VD_Constraint_Set** class ([7.4.2.20](#));
- **Local_Enumerated_VD_Constraint_Set** class ([7.4.2.21](#));
- **Local_Enumerated_VD_Subset_Constraint_Set** class ([7.7.2.8](#));
- **Permissible_Value_Constraint_Set** class ([7.4.2.22](#)).

Any registry implementation shall provide a mechanism to enforce these constraints.

7 Data_Specification package

7.1 Overview of the Data_Specification package

This package extends the Basic_and_Core package specified in ISO/IEC 11179-3:2023, Clause 6 by sub-classing classes in that package, both **Item** (ISO/IEC 11179-3:2023, 6.4.2.1) and **Concept** (ISO/IEC 11179-3:2023, 6.4.2.2).

The Data Specification package consists of eight metamodel regions, or views of the model:

- the High-level Data Specification metamodel region ([7.2](#));

- the Data Element Concept metamodel region (7.3);
- the Conceptual and Value Domain metamodel region (7.4);
- the Measurement metamodel region (7.5);
- the Data Element metamodel region (7.6);
- the Value Domain Subset metamodel region (7.7);
- the Conceptual Domain Subset metamodel region (7.8);
- the Composite Data Element and Datatypes metamodel region (7.9).

7.2 High-level Data Specification metamodel region

7.2.1 Overview of the high-level metamodel region

A high-level overview of the metamodel for this region can be found in [Figure 3](#). It shows four classes: **Conceptual_Domain** (7.2.2.2), **Value_Domain** (7.2.2.3), **Data_Element** (7.2.2.4) and **Data_Element_Concept** (7.2.2.5). The Figure also shows four associations among the four classes: **value_domain_meaning** (7.2.3.1), **data_element_domain** (7.2.3.2), **data_element_meaning** (7.2.3.3) and **data_element_concept_domain** (7.2.3.4).

The following text describes the classes and associations shown in the Figure. It also describes a constraint on the high level metamodel not visible in the UML diagram. More detailed descriptions, e.g. of the class attributes, follow in subsequent subclauses.

[Figure 3](#) can be partitioned into two horizontal parts, one upper part comprised of **Data_Element_Concept** and **Conceptual_Domain** and a second lower part comprised of **Data_Element** and **Value_Domain**. This view effectively splits the metamodel between a conceptual (or semantic) level (at the top) and a representational level (below). The representational level describes the information artifacts (in contrast to the semantic constructs of the upper level).

This high-level metamodel omits many details, e.g. attributes and some associations, in the interest of clarity of exposition. For a complete characterization of the metamodel see [7.3](#) through [7.9](#). A partial consolidated detailed metamodel is presented in [Annex C](#).

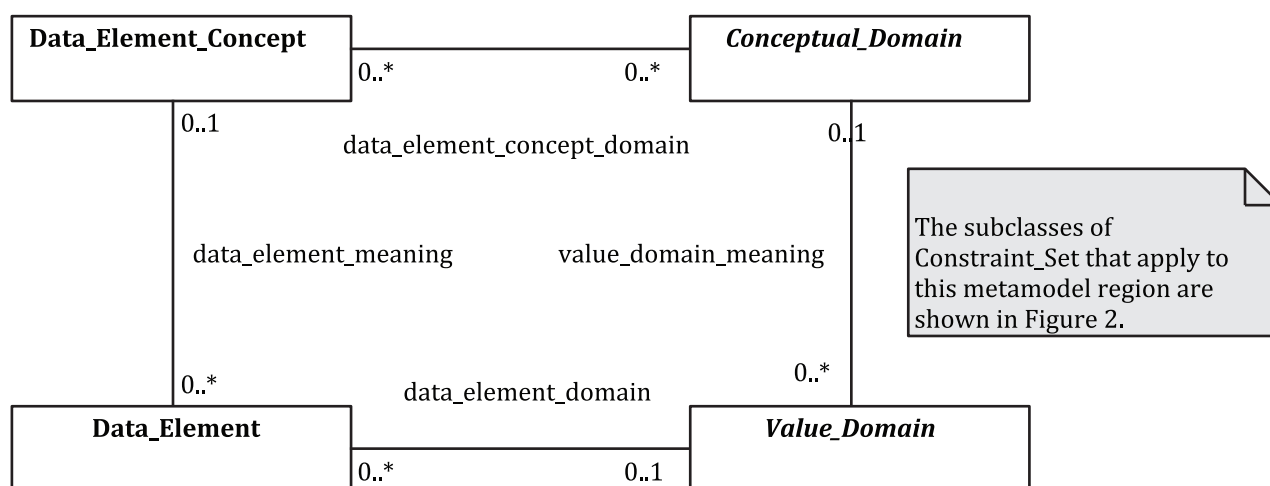


Figure 3 — High-level Data Specification metamodel

ISO/IEC 11179-3:2003^[4] required a data element concept to be associated with exactly one conceptual domain. This association was relaxed in ISO/IEC 11179-3:2013^[5] and in this document because of feedback from organizations using ISO/IEC 11179-3:2003:

- The lower bound has been relaxed from one to zero to allow a data element concept to be recorded without an associated conceptual domain.
- The upper bound has been increased from one to many to avoid a proliferation of similar data element concepts. For example, it is possible for the data element concept “Person’s sex” in one context to be associated with a conceptual domain of { male , female } and in another context with a conceptual domain of { male , female , unknown , not disclosed }. ISO/IEC 11179-3:2003 required that separate data element concepts be specified. ISO/IEC 11179-3:2013 and this document allow an organization to choose whether to specify separate data element concepts or to share a single data element concept. Whichever approach is chosen for data element concepts, it is best practice for separate data elements to be defined and associated with the appropriate value domains.

A registration authority can reapply the constraints which have been relaxed. See [7.2.4.2](#).

7.2.2 Classes in the High-level Data Specification metamodel region

7.2.2.1 Overview

The classes shown in [Figure 3](#) are described in this subclause, starting with **Conceptual_Domain** ([7.2.2.2](#)) and proceeding clockwise around the figure.

7.2.2.2 Conceptual_Domain class

7.2.2.2.1 Direct superclass

Conceptual_Domain is a subclass of **Concept** (specified in ISO/IEC 11179-3:2023, 6.4.2.2). **Concept** in turn is a subclass of **Item** (specified in ISO/IEC 11179-3:2023, 6.4.2.1), allowing instances to be identified, registered, administered, named, defined and classified.

7.2.2.2.2 Description of Conceptual_Domain

Conceptual_Domain is a class each instance of which models a conceptual domain, a concept whose meaning is expressed as an enumerated set, a description of subordinate concepts or both, which are value meanings.

For example, one possible conceptual domain is “countries of the world”. It can, for example, be associated with two value domains: “three letter country codes” and “full country names”. The conceptual domain can be used in several data element concepts, e.g. “person’s country of residence”, “person’s country of birth”, “person’s country of citizenship”.

A conceptual domain can facilitate the mapping of equivalent values of two or more value domains that share the conceptual domain.

Conceptual_Domain is further described in [7.4.2.1](#).

7.2.2.2.3 Associations of Conceptual_Domain

As a subclass of **Concept**, **Conceptual_Domain** inherits **Concept**’s associations (ISO/IEC 11179-3:2023, 6.4.2.2.3). **Conceptual_Domain** has the following additional associations:

- **data_element_concept_domain** ([7.2.3.4](#));
- **value_domain_meaning** ([7.2.3.1](#)).

More associations of **Conceptual_Domain** are described in [7.4.2.1.3](#).

7.2.2.2.4 Attributes of Conceptual_Domain

Attributes of **Conceptual_Domain** are described in [7.4.2.1.4](#).

7.2.2.3 Value_Domain class

7.2.2.3.1 Direct superclass

Value_Domain is a subclass of **Item** (specified in ISO/IEC 11179-3:2023, 6.4.2.1), allowing instances to be identified, registered, administered, named, defined and classified.

7.2.2.3.2 Description of Value_Domain

Value_Domain is a class each instance of which models a value domain, a collection of permissible values. A value domain provides representation but has no implication as to what data element concept the values are associated with, nor what the values mean. The binding of permissible values to their corresponding value meanings is described in [7.4.3.2.2](#).

Instances of the **Value_Domain** class are each associated with zero or one instance of the **Conceptual_Domain** class ([7.2.2.2](#)) through instances of the association **value_domain_meaning** ([7.2.3.1](#)). Through this association, a value domain provides a representation for the conceptual domain and the conceptual domain provides meaning for the value domain. Although the association is shown as optional in both directions, without such an association, a value domain has no meaning and a conceptual domain has no representation.

An example of a conceptual domain and a set of value domains is ISO 3166^[1] Codes for the representation of names of countries. For instance, ISO 3166 describes the set of seven value domains:

- short name in English;
- official name in English;
- short name in French;
- official name in French;
- alpha-2 code;
- alpha-3 code;
- numeric code.

Additional examples of value domains are:

- Sex, which contains two designations (permissible values), M -> Male and F-> Female;
- Parent, which contains two designations (permissible values), M -> Mother and F -> Father.

NOTE These two value domains are defined over the same set of values, but they are mapped to separate conceptual domains.

Value_Domain is further described in [7.4.2.8](#).

Value domains may be reused for multiple data elements. See the discussion of countries of the world in [7.2.2.2.2](#).

7.2.2.3.3 Associations of Value_Domain

As a subclass of **Item**, **Value_Domain** inherits **Item**'s associations (ISO/IEC 11179-3:2023, 6.4.2.1.2). **Value_Domain** has the following additional associations:

- **value_domain_meaning** ([7.2.3.1](#));

— **data_element_domain** ([7.2.3.2](#)).

7.2.2.3.4 Attributes of Value_Domain

Attributes of **Value_Domain** are described in [7.4.2.8.4](#).

7.2.2.4 Data_Element class

7.2.2.4.1 Direct superclass

Data_Element is a subclass of **Item** (specified in ISO/IEC 11179-3:2023, 6.4.2.1), allowing instances to be identified, registered, administered, named, defined and classified.

7.2.2.4.2 Description of 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 database managed using the SQL database language, a field in a record or form, an XML element, the attribute of a Java class or a variable in a program. The description of data elements was the original major purpose of the metadata registries described in ISO/IEC 11179.

Data_Element is further described in [7.6.2.1](#).

7.2.2.4.3 Associations of Data_Element

As a subclass of **Item**, **Data_Element** inherits **Item**'s associations (ISO/IEC 11179-3:2023, 6.4.2.1.2). **Data_Element** has the following additional associations:

- **data_element_meaning** ([7.2.3.3](#));
- **data_element_domain** ([7.2.3.2](#)).

Additional associations of **Data_Element** are described in [7.6.2.1.3](#).

7.2.2.4.4 Attributes of Data_Element

Attributes of **Data_Element** are described in [7.6.2.1.4](#).

7.2.2.5 Data_Element_Concept class

7.2.2.5.1 Direct superclass

Data_Element_Concept is a subclass of **Concept** (specified in ISO/IEC 11179-3:2023, 6.4.2.2). **Concept** in turn is a subclass of **Item** (specified in ISO/IEC 11179-3:2023, 6.4.2.1), allowing instances to be identified, registered, administered, named, defined and classified.

7.2.2.5.2 Description of Data_Element_Concept

Data_Element_Concept is a class each instance of which models a data element concept, a concept that is an association of a property with an object class. A data element concept is a concept that can be represented in the form of a data element, described independently of any particular representation.

A data element concept is a usage of a conceptual domain, e.g. "person's country of residence" vs. "country", which effectively narrows the meaning of the conceptual domain.

A data element concept is an abstraction of one or more data elements. Each data element addresses issues of concrete representation, e.g. codes, measurement units, etc. A data element concept may be represented by multiple data elements, which may vary in their value domains.

A data element concept can facilitate the mapping of equivalent values of two or more data elements that share the data element concept.

Data_Element_Concept is further described in [7.3.2.3](#).

7.2.2.5.3 Associations of Data_Element_Concept

As a subclass of **Concept**, **Data_Element_Concept** inherits **Concept**'s associations (ISO/IEC 11179-3:2023, 6.4.2.2.3). **Data_Element_Concept** has the following additional associations:

- **data_element_concept_domain** ([7.2.3.4](#));
- **data_element_meaning** ([7.2.3.3](#)).

Other associations of **Data_Element_Concept** are described in [7.3.2.3.3](#).

7.2.2.5.4 Attributes of Data_Element_Concept

As a subclass of **Concept**, **Data_Element_Concept** inherits **Concept**'s attributes (ISO/IEC 11179-3:2023, 6.4.2.2.4).

7.2.3 Associations of the High-Level Data Specification metamodel region

7.2.3.1 value_domain_meaning association

The **value_domain_meaning** association records the binding of zero, one or more instances of the **Value_Domain** class ([7.2.2.3](#)) to zero or one instances of the **Conceptual_Domain** class ([7.2.2.2](#)). The association records the value domains which provide representation to the conceptual domain, which in turn provides meaning to the value domains.

It is best practice for a value domain to be associated with exactly one conceptual domain. A registration authority can enforce this using the **Value_Domain_Constraint_Set.VD_shall_have_exactly_one_CD** constraint ([7.4.2.18.4](#)).

A value domain is a collection of permissible values which provide values for associated value meanings. The existence of a **value_domain_meaning** association between an instance of the **Conceptual_Domain** class and an instance of the **Value_Domain** class implies the existence of associations between the corresponding individual value meanings and permissible values. These associations are recorded in this metamodel as **Permissible_Values** ([7.4.2.12](#)) for enumerated value domains. For described value domains, the associations are implicit.

NOTE In this metamodel, value domains are constrained to have a unique set of meanings (the associated conceptual domain), i.e. a value domain is a function from permissible values to value meanings. If for some reason one wanted to reuse a value domain (and the associated values, e.g. a code set) for more than one meaning (e.g. see the additional examples in [7.2.2.3](#)), one is forced to create another value domain and another set of permissible values. This constraint is enforced so that within a value domain, one can unambiguously determine the value meanings (in the conceptual domain) for the permissible values (in the value domain) associated with a data element. See discussion under Constraints in [7.2.4.1](#).

7.2.3.2 data_element_domain association

The **data_element_domain** association binds zero, one or more instances of the **Data_Element** class ([7.2.2.4](#)) to zero or one instances of the **Value_Domain** class ([7.2.2.3](#)) which specifies the permissible values which may be stored in the data element. The value domain provides a domain of permissible values for the data element, which in turn provides usages of the value domain.

It is best practice for a data element to be associated with exactly one value domain. A registration authority can enforce this using the constraint **Data_Element_Constraint_Set.DE_shall_have_exactly_one_VD** (7.6.2.7.4).

7.2.3.3 data_element_meaning association

The **data_element_meaning** association binds zero, one or more instances of the **Data_Element** class (7.2.2.4) to zero or one instance of the **Data_Element_Concept** class (7.2.2.5). The data element concept provides meaning to the data elements, which in turn provide representation to the data element concept.

It is best practice for a data element to be associated with exactly one data element concept. A registration authority can enforce this using the **Data_Element_Constraint_Set.DE_shall_have_exactly_one_DEC** constraint (7.6.2.7.4).

7.2.3.4 data_element_concept_domain association

The **data_element_concept_domain** association binds zero, one or more instances of the **Data_Element_Concept** class (7.2.2.5) to zero, one or more instances of the **Conceptual_Domain** class (7.2.2.2). The conceptual domains provide the domain for each data element concept, which in turn provide usages for the conceptual domain.

It is best practice for a data element concept to be associated with exactly one conceptual domain. A registration authority can enforce this using the constraints **Data_Element_Concept_Constraint_Set.DEC_shall_have_at_least_one_CD** and **Data_Element_Concept_Constraint_Set.DEC_shall_have_at_most_one_CD** (7.3.2.5.4).

The **data_element_concept_domain** association narrows the scope (meaning) of a conceptual domain to that of the data element concept.

EXAMPLE “person’s country of birth” (data element concept) vs. “country” (conceptual domain).

NOTE ISO/IEC 11179-3:2003^[4] required a data element concept to be associated with exactly one conceptual domain. This association was relaxed in ISO/IEC 11179-3:2013^[5] and in this document because of feedback from organizations using ISO/IEC 11179-3:2003. The lower bound has been relaxed from one to zero to allow a data element concept to be recorded without an associated conceptual domain. The upper bound has been increased from one to many, to avoid a proliferation of similar data element concepts. For example, it is possible for the data element concept: “Person’s marital status” in one context to be associated with a conceptual domain of { single , married }, and in another context with a conceptual domain of { single , married , divorced , widowed }. ISO/IEC 11179-3:2003 required that separate data element concepts be specified to support this. ISO/IEC 11179-3:2013 and this document allow an organization to choose whether to specify separate data element concepts or to share a single data element concept.

Even if a data element concept is associated with multiple conceptual domains, separate data elements should still be defined and associated with the appropriate value domains.

7.2.4 Constraints of the High Level metamodel region

7.2.4.1 Equality of mappings from data element to conceptual domain

There are two paths in the metamodel in Figure 3 from the **Data_Element** class to the **Conceptual_Domain** class. One can either proceed clockwise from **Data_Element** class via the **data_element_meaning** association to **Data_Element_Concept** class and then via **data_element_concept_domain** association to the **Conceptual_Domain** class; or alternatively, one can proceed counterclockwise from the **Data_Element** class via the **data_element_domain** association to the **Value_Domain** class and then via the **value_domain_meaning** association to the **Conceptual_Domain** class.

CONSTRAINT: It shall be the case that starting from a specific instance of the **Data_Element** class that the instance of the **Conceptual_Domain** class reached via the **Value_Domain** shall be one of those reached via the **Data_Element_Concept**.