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

**Part 3:
Registry metamodel and basic attributes**

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



Reference number
ISO/IEC 11179-3:2013(E)

AVA Documents
Document Sharing Hub
© ISO/IEC 2013



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2013

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword	xiii
Introduction.....	xv
1 Scope	1
1.1 Scope – Structure of a metadata registry	1
1.2 Scope – Basic attributes of metadata items	1
2 Normative references.....	1
3 Terms, definitions and abbreviated terms	1
3.1 Terms and definitions of metamodel constructs used in this part of ISO/IEC 11179	2
3.2 Terms for concepts used in this part of ISO/IEC 11179	4
3.3 Abbreviated terms	20
4 Conformance	21
4.1 Overview of conformance	21
4.2 Degree of conformance	21
4.2.1 General	21
4.2.2 Strictly conforming implementations	22
4.2.3 Conforming implementations	22
4.3 Conformance by clause	22
4.4 Registry conformance.....	23
4.4.1 Overview.....	23
4.4.2 Standard profiles for edition 3 registries	23
4.5 Obligation	23
4.6 Implementation conformance statement (ICS).....	23
4.7 Roles and responsibilities for registration	24
5 Structure of a metadata registry	24
5.1 Metamodel for a metadata registry	24
5.2 Application of the metamodel	24
5.3 Specification of the metamodel	25
5.3.1 Terminology used in specifying the metamodel.....	25
5.3.2 Choice of fonts	25
5.3.3 Use of UML Packages	25
5.3.4 Package dependencies	26
5.3.5 Use of UML Class diagrams and textual description	27
5.4 Types, instances and values	27
5.5 Types of items in an ISO/IEC 11179 metadata registry	28
5.5.1 Overview of types of items	28
5.5.2 Rules for types of items	29
5.6 Extensibility	31
5.7 Date references.....	31
6 Basic package.....	31
6.1 Overview of Basic package	31
6.2 Basic Types metamodel region	31
6.2.1 Overview of Basic Types	31
6.2.2 Boolean datatype.....	31
6.2.3 Date datatype	32
6.2.4 Datetime datatype.....	32
6.2.5 Integer datatype	32
6.2.6 Natural_Range datatype	32
6.2.7 Notation datatype	32
6.2.8 Phone_Number datatype	32

6.2.9	Postal_Address datatype	32
6.2.10	Sign datatype	33
6.2.11	String datatype	33
6.2.12	Text datatype	33
6.2.13	Value datatype	33
6.3	Basic Classes metamodel region	33
6.3.1	Overview of Basic Classes	33
6.3.2	Contact class	34
6.3.3	Document_Type class	35
6.3.4	Individual class	35
6.3.5	Language_Identification class	37
6.3.6	Organization class	39
6.3.7	Reference_Document class	40
6.3.8	Registration_Authority_Identifier class	42
6.3.9	Role class	43
7	Identification, Designation and Definition package	45
7.1	Overview of this package	45
7.2	Identification metamodel region	45
7.2.1	Overview	45
7.2.2	Classes in the Identification metamodel region	46
7.2.3	Associations in the Identification metamodel region	51
7.3	Designation and Definition metamodel region	52
7.3.1	Overview	52
7.3.2	Classes in the Designation and Definition metamodel region	53
7.3.3	Association Classes in the Designation and Definition metamodel region	59
7.3.4	Associations in the Designation and Definition metamodel region	60
8	Registration package	62
8.1	Registration metamodel region	62
8.1.1	Overview	62
8.1.2	Classes in the Registration region	62
8.1.3	Classes referenced from the Basic package	75
8.1.4	Classes referenced from the Identification, Designation and Definition package	75
8.1.5	Association Classes in the Registration region	76
8.1.6	Associations in the Registration region	76
9	Concepts package	78
9.1	Concepts metamodel region	78
9.1.1	Overview	78
9.1.2	Classes in the Concepts metamodel region	79
9.1.3	Associations of the Concepts metamodel region	84
9.2	Classification metamodel region	87
9.2.1	Overview	87
9.2.2	Classes in the Classification metamodel region	88
9.2.3	Associations Classes in the Classification metamodel region	89
9.2.4	Associations in the Classification metamodel region	89
10	Binary Relations package	90
10.1	Binary Relations metamodel region	90
10.1.1	Overview	90
10.1.2	Classes in the Binary_Relations metamodel region	90
11	Data Description package	93
11.1	High-level Data Description metamodel region	93
11.1.1	Overview	93
11.1.2	Classes of High-level Data Description metamodel	93
11.1.3	Associations of the High Level Data Description metamodel	96
11.1.4	Constraints of the High Level Metamodel	97
11.2	Data Element Concept metamodel region	97
11.2.1	Overview	97
11.2.2	Classes in the Data_Element_Concept region	98

11.2.3	Associations in the Data_Element_Concept region	99
11.3	Conceptual and Value_Domain metamodel region	100
11.3.1	Overview	100
11.3.2	Classes in the Conceptual and Value_Domain region	102
11.3.3	Associations in the Conceptual and Value_Domain region	110
11.3.4	Additional Constraints of the Conceptual and Value_Domain region	112
11.4	Measurement metamodel region	114
11.4.1	Overview	114
11.4.2	Classes in the Measurement region	114
11.4.3	Associations in the Measurement region	117
11.5	Data_Element metamodel region	118
11.5.1	Overview	118
11.5.2	Classes in the Data_Element Region	118
11.5.3	Associations in the Data_Element region	121
11.6	Consolidated Data Description Metamodel	123
11.7	Types of Concepts in the Data Description Metamodel	124
12	Basic attributes	124
12.1	Use of basic attributes	124
12.2	Common attributes	125
12.2.1	Identifying	125
12.2.2	Naming	125
12.2.3	Definitional	126
12.2.4	Administrative	126
12.2.5	Relational	126
12.3	Attributes specific to Data_Element_Concepts	127
12.4	Attributes specific to Data_Elements	127
12.5	Attributes specific to Conceptual_Domains	127
12.6	Attributes specific to Value_Domains	127
12.7	Attributes specific to Permissible_Values	128
12.8	Attributes specific to Value_Meanings	128
Annex A (normative)	Alphabetical list of terms and designations	129
Annex B (normative)	Consolidated Class Hierarchy	137
Annex C (informative)	Mapping the ISO/IEC 11179-3:1994 basic attributes to the ISO/IEC 11179-3:2011 metamodel and basic attributes	138
C.1	Introduction	138
C.1.1	Overview of Basic Attributes from ISO/IEC 11179-3:1994	138
C.1.2	Description of Table Structures in this Annex	139
C.2	Mapping the Basic Attributes	141
C.2.1	Common Identifying attributes	141
C.2.2	Common Naming attributes	143
C.2.3	Common Definitional attributes	147
C.2.4	Common Administrative attributes	148
C.2.5	Common Relational attributes	150
C.2.6	Attributes specific to Data_Element_Concepts	154
C.2.7	Attributes specific to Data_Elements	157
C.2.8	Attributes specific to Conceptual_Domains	164
C.2.9	Attributes specific to Value_Domains	165
C.2.10	Attributes specific to Permissible_Values	166
C.2.11	Attributes specific to Value_Meanings	167
Annex D (informative)	Mapping the ISO/IEC 11179-3:2003 metamodel to the ISO/IEC 11179-3:2011 metamodel	169
D.1	Introduction	169
D.2	Mapping the Edition 2 Administration and Identification Region	169
D.2.1	Administered_Item	169
D.2.2	Administration_Record	169
D.2.3	Contact	170
D.2.4	Item_Identifier	170

D.2.5	Language_Identification	170
D.2.6	Organization	171
D.2.7	Reference_Document	171
D.2.8	Registrar	171
D.2.9	Registration_Authority	171
D.2.10	Registration_Authority_Identifier	172
D.2.11	Stewardship	172
D.2.12	Submission	172
D.3	Mapping the Edition 2 Naming and Definition Region	173
D.3.1	Context (for Administered_Item)	173
D.3.2	Terminological_Entry	173
D.3.3	Language_Section	173
D.3.4	Definition (of Administered_Item)	174
D.3.5	Designation (of Administered_Item)	174
D.4	Mapping the Edition 2 Classification Region	174
D.4.1	Classification_Scheme	174
D.4.2	Classification_Scheme_Item	175
D.4.3	Classification_Scheme_Item_Relationship	175
D.5	Mapping the Edition 2 Data_Element_Concept Region	175
D.5.1	Object_Class	175
D.5.2	Property	175
D.5.3	Data_Element_Concept	176
D.5.4	Concept_Relationship	176
D.6	Mapping the Edition 2 Conceptual and Value Domain Region	176
D.6.1	Conceptual_Domain	176
D.6.2	Enumerated_Conceptual_Domain	177
D.6.3	Value_Meaning	177
D.6.4	Non-enumerated_Conceptual_Domain	177
D.6.5	Value_Domain	178
D.6.6	Enumerated_Value_Domain	178
D.6.7	Permissible_Value	178
D.6.8	Value	179
D.6.9	Non-enumerated_Value_Domain	179
D.6.10	Datatype	179
D.6.11	Unit_of_Measure	179
D.7	Mapping the Edition 2 Data_Element Region	180
D.7.1	Data_Element	180
D.7.2	Representation_Class	180
D.7.3	Data_Element_Example	181
D.7.4	Derivation_Rule	181
D.7.5	Data_Element_Derivation	181
Annex E (informative)	Concept System Examples	182
E.1	Concept System Metamodels	182
E.2	SKOS Example	183
E.2.1	SKOS Metamodel	183
E.2.2	SKOS Example Thesaurus	184
E.2.3	Example Value Domain References	185
E.3	ORM Example	187
E.3.1	ORM Metamodel	187
E.3.2	Car Registration Model	189
E.4	OWL Example	194
E.4.1	OWL Metamodel	194
E.4.2	Car Registration Ontology	202
E.5	CLIF Example	216
E.5.1	CL Metamodel	216
E.5.2	CLIF Units Example from ISO/IEC 19763-3	217
Annex F (informative)	Representation Class as a Concept System	221
F.1	Introduction	221
F.2	Description of Representation Class	221

F.3	Implementation of Representation Class as a Concept_System	222
Annex G	(informative) Comparison for Conformance Levels across Editions of this part of ISO/IEC 11179	223
G.1	Introduction.....	223
G.2	Conformance Levels for Edition 2 Level 2.....	223
G.3	Conformance Levels for Edition 2 Level 1 and Edition 1	223
Annex H	(Normative) Standard Conformance Profiles for this part of ISO/IEC 11179	224
H.1	Introduction.....	224
H.2	Profile for Concept Systems Registry.....	224
H.3	Profile for Extended Concept Systems Registry	224
H.4	Profile for Metadata Registry	224
H.5	Profile for Extended Metadata Registry	224
	Bibliography.....	225

Table of Figures

Figure 1 — Package dependencies	26
Figure 2 — Types of items	28
Figure 3 — Basic types metamodel region	31
Figure 4 — Basic classes metamodel region	33
Figure 5 — Identification metamodel region.....	45
Figure 6 — Designation and Definition metamodel region.....	53
Figure 7 — Registration metamodel region.....	64
Figure 8 — Concepts metamodel region.....	78
Figure 9 — Classification metamodel region.....	88
Figure 10 — Binary Relations metamodel region.....	90
Figure 11 — High-level Data Description metamodel	93
Figure 12 — Data_Element_Concept metamodel region	98
Figure 13 — Conceptual and value domain metamodel region	101
Figure 14 — Measurement metamodel region.....	114
Figure 15 — Data_Element metamodel region	118
Figure 16 — Consolidated Data Description metamodel	123
Figure 17 — Types of Concepts in the Data Description package	124
Figure 18 — Consolidated Class Hierarchy	137

Figure 19 — Basic Attributes of Data elements.....	138
Figure 20— Car Registration Model in ORM.....	189
Figure 21 — Car Registration Ontology.....	202

Table of Tables

Table 1 – Rules for Types of Items.....	29
Table 2 – Rules for Types of Items as a Decision Table	30
Table 3 – Comparison of Designation to Scoped_Identifier	45
Table 4 – Examples of binary relations and their characterization.....	90
Table 5 – Template for attribute mapping.....	139
Table 6 – Attribute mapping for ‘identifier’	141
Table 7 – Attribute mapping for ‘Registration Authority’	142
Table 8 – Attribute mapping for ‘Version’.....	142
Table 9 – Attribute mapping for ‘Name’	143
Table 10 – Attribute mapping for ‘Synonymous name’.....	143
Table 11 – Attribute mapping for ‘designation language’	144
Table 12 – Attribute mapping for ‘Context name’	144
Table 13 – Attribute mapping for ‘Context identifier’.....	145
Table 14 – Attribute mapping for ‘Context description’	146
Table 15 – Attribute mapping for ‘Definition’.....	147
Table 16 – Attribute mapping for ‘Definition language’.....	147
Table 17 – Attribute mapping for ‘Definition source reference’	148
Table 18 – Attribute mapping for ‘Comments’	148
Table 19 – Attribute mapping for ‘Registration status’.....	148
Table 20 – Attribute mapping for ‘Responsible organization’	149
Table 21 – Attribute mapping for ‘Submitting organization’	150
Table 22 – Attribute mapping for ‘Classification scheme name’	150

Table 23 – Attribute mapping for ‘Classification scheme identifier’	151
Table 24 – Attribute mapping for ‘Classification scheme item value’	152
Table 25 – Attribute mapping for ‘Related metadata reference’	153
Table 26 – Attribute mapping for ‘Type of relationship’	154
Table 27 – Attribute mapping for ‘Object class name’	154
Table 28 – Attribute mapping for ‘Object class identifier’	155
Table 29 – Attribute mapping for ‘Property name’	156
Table 30 – Attribute mapping for ‘Property identifier’	156
Table 31 – Attribute mapping for ‘Representation category’	157
Table 32 – Attribute mapping for ‘Representation class’	157
Table 33 – Attribute mapping for ‘Value domain name’	158
Table 34 – Attribute mapping for ‘Value domain identifier’	159
Table 35 – Attribute mapping for ‘Datatype name’	159
Table 36 – Attribute mapping for ‘Datatype scheme reference’	160
Table 37 – Attribute mapping for ‘Maximum size’	161
Table 38 – Attribute mapping for ‘Minimum size’	162
Table 39 – Attribute mapping for ‘Layout of representation’	163
Table 40 – Attribute mapping for ‘Permissible data element values’	164
Table 41 – Attribute mapping for ‘Dimensionality’	164
Table 42 – Attribute mapping for ‘Datatype name’	165
Table 43 – Attribute mapping for ‘Datatype scheme reference’	165
Table 44 – Attribute mapping for ‘Unit of measure name’	166
Table 45 – Attribute mapping for ‘Value’	166
Table 46 – Attribute mapping for ‘Permissible value begin date’	167
Table 47 – Attribute mapping for ‘Permissible value end date’	167
Table 48 – Attribute mapping for ‘Value meaning description’	167
Table 49 – Attribute mapping for ‘Value meaning identifier’	168
Table 50 – Attribute mapping for ‘Value meaning begin date’	168
Table 51 – Attribute mapping for ‘Value meaning end date’	168
Table 52 – Mapping Edition 2 Administered_Item to Edition 3	169

Table 53 – Mapping Edition 2 Administration_Record to Edition 3	169
Table 54 – Mapping Edition 2 Contact to Edition 3	170
Table 55 – Mapping Edition 2 Item_Identifier to Edition 3	170
Table 56 – Mapping Edition 2 Language_Identification to Edition 3	170
Table 57 – Mapping Edition 2 Organization to Edition 3	171
Table 58 – Mapping Edition 2 Reference_Document to Edition 3	171
Table 59 – Mapping Edition 2 Registrar to Edition 3	171
Table 60 – Mapping Edition 2 Registration_Authority to Edition 3	171
Table 61 – Mapping Edition 2 Registration_Authority_Identifier to Edition 3	172
Table 62 – Mapping Edition 2 Stewardship to Edition 3	172
Table 63 – Mapping Edition 2 Submission to Edition 3	172
Table 64 – Mapping Edition 2 Context to Edition 3	173
Table 65 – Mapping Edition 2 Terminological_Entry to Edition 3	173
Table 66 – Mapping Edition 2 Language_Section to Edition 3	173
Table 67 – Mapping Edition 2 Definition (of Administered_Item) to Edition 3	174
Table 68 – Mapping Edition 2 Designation (of Administered_Item) to Edition 3	174
Table 69 – Mapping Edition 2 Classification_Scheme to Edition 3	174
Table 70 – Mapping Edition 2 Classification_Scheme_Item to Edition 3	175
Table 71 – Mapping Edition 2 Classification_Scheme_Item_Relationship to Edition 3	175
Table 72 – Mapping Edition 2 Object_Class to Edition 3	175
Table 73 – Mapping Edition 2 Property to Edition 3	175
Table 74 – Mapping Edition 2 Property to Edition 3	176
Table 75 – Mapping Edition 2 Property to Edition 3	176
Table 76 – Mapping Edition 2 Conceptual_Domain to Edition 3	176
Table 77 – Mapping Edition 2 Enumerated_Conceptual_Domain to Edition 3	177
Table 78 – Mapping Edition 2 Value_Meaning to Edition 3	177
Table 79 – Mapping Edition 2 Non-enumerated_Conceptual_Domain to Edition 3	177
Table 80 – Mapping Edition 2 Value_Domain to Edition 3	178
Table 81 – Mapping Edition 2 Enumerated_Value_Domain to Edition 3	178
Table 82 – Mapping Edition 2 Permissible_Value to Edition 3	178

Table 83 – Mapping Edition 2 Value to Edition 3	179
Table 84 – Mapping Edition 2 Non-enumerated_Value_Domain to Edition 3.....	179
Table 85 – Mapping Edition 2 Datatype to Edition 3	179
Table 86 – Mapping Edition 2 Unit_of_Measure to Edition 3	179
Table 87 – Mapping Edition 2 Data_Element to Edition 3.....	180
Table 88 – Mapping Edition 2 Representation_Class to Edition 3	180
Table 89 – Mapping Edition 2 Data_Element_Example to Edition 3	181
Table 90 – Mapping Edition 2 Derivation_Rule to Edition 3.....	181
Table 91 – Mapping Edition 2 Data_Element_Derivation to Edition 3	181
Table 92 – Correspondences of ISO/IEC 11179-3 concept system metamodel to selected notations	182
Table 93 – SKOS-CORE as an ISO/IEC 11179 Concept System	183
Table 94 – SKOS relations as ISO/IEC 11179 Binary Relations	184
Table 95 – SKOS Thesaurus Example – ISO/IEC 11179 Concept System	184
Table 96 – SKOS Thesaurus Example – ISO/IEC 11179 Concepts.....	185
Table 97 – SKOS Thesaurus Example – ISO/IEC 11179 Links.....	185
Table 98 – SKOS Thesaurus Example – ISO/IEC 11179 Conceptual Domains	186
Table 99 – SKOS Thesaurus Example – ISO/IEC 11179 Value Domains	186
Table 100 – ORM as an ISO/IEC 11179 Concept System	187
Table 101 – ORM Relations as ISO/IEC 11179 Binary Relations.....	188
Table 102 – ORM Roles as ISO/IEC 11179 Relation Roles	188
Table 103 – Car Registration Model in ORM – ISO/IEC 11179 Concept System	191
Table 104 – Car Registration Model in ORM – ISO/IEC 11179 Concepts.....	191
Table 105 – Car Registration Model in ORM – ISO/IEC 11179 Binary Relations.....	192
Table 106 – Car Registration Model in ORM – ISO/IEC 11179 Links	193
Table 107 – OWL constructs with directly corresponding ISO/IEC 11179-3 metamodel elements.....	194
Table 108 – OWL built-in constructs described in OWL metamodel.....	195
Table 109 – OWL as an ISO/IEC 11179 Concept System.....	195
Table 110 – OWL Concepts as ISO/IEC 11179 Concepts.....	196
Table 111 – OWL Binary Relations as ISO/IEC 11179 Binary Relations	197
Table 112 – OWL Relations (except Binary Relations) as ISO/IEC 11179 Relations.....	198

Table 113 – OWL Constructs as ISO/IEC 11179 Relation Roles.....	198
Table 114 – OWL Constructs as ISO/IEC 11179 Links.....	200
Table 115 – Car Registration Model in OWL – ISO/IEC 11179 Concept System.....	207
Table 116 – Car Registration Model in OWL – ISO/IEC 11179 Concepts	207
Table 117 – Car Registration Model in OWL – ISO/IEC 11179 Binary Relations	208
Table 118 – Car Registration Model in OWL – ISO/IEC 11179 Relation Roles.....	208
Table 119 – Car Registration Model in OWL – ISO/IEC 11179 Links	209
Table 120 – Car Registration Model in OWL – ISO/IEC 11179 Assertions.....	214
Table 121 – CL Metamodel – ISO/IEC 11179 Concept System	216
Table 122 – CL Metamodel – ISO/IEC 11179 Binary Relations.....	216
Table 123 – CLIF Units Example – ISO/IEC 11179 Concept System.....	217
Table 124 – CLIF Units Example – ISO/IEC 11179 Concepts	218
Table 125 – CLIF Units Example – ISO/IEC 11179 Binary Relations	218
Table 126 – CLIF Units Example – ISO/IEC 11179 Relations Roles.....	219
Table 127 – CLIF Units Example – ISO/IEC 11179 Links	219
Table 128 – CLIF Units Example – ISO/IEC 11179 Assertions	220
Table 129 – Comparison for Conformance Levels across Editions of this part of ISO/IEC 11179	223

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. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 11179-3 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 32, *Data management and interchange*.

This third edition cancels and replaces the second edition (ISO/IEC 11179-3:2003) and ISO/IEC 11179-3:2003/Cor 1:2004. The changes are as follows:

Edition 3 of this part of ISO/IEC 11179 includes several enhancements to Edition 2, both in terms of the presentation of the metamodel, and its capabilities, as follows:

From a presentation perspective, these include:

- use of UML 2.4.1 instead of UML 1.4 to describe the metamodel;
- use of UML packages to show dependencies between various regions of the metamodel. (See 5.3.3 and 5.3.4.)

From a capability perspective, these include:

- introduction of different types of metadata items (see 5.5);
- support for registration of Concept Systems (see 9.1);
- finer-grained conformance options (see 4.3).

ISO/IEC 11179 consists of the following parts, under the general title *Information technology — Metadata registries (MDR)*:

- *Part 1: Framework*
- *Part 2: Classification*
- *Part 3: Registry metamodel and basic attributes*
- *Part 4: Formulation of data definitions*

- *Part 5: Naming and identification principles*
- *Part 6: Registration*

Introduction

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

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

When considering data and metadata, it is important to distinguish between types of data/metadata, and instances of these types. Clause 5 through 11 of this part of ISO/IEC 11179 specify the types of metadata objects that form the structure of a metadata registry. A metadata registry will be populated with instances of these metadata objects (metadata items), which in turn define types of data, e.g. in an application database. In other words, instances of metadata specify types of application level data. In turn, the application database will be populated by the real world data as instances of those defined datatypes.

NOTE ISO/IEC 10027:1990, *Information technology — Information resource dictionary system (IRDS) Framework* and ISO/IEC TR 10032:2003, *Information technology — Reference model for data management* explain the concepts of different levels of modelling.

In this part of ISO/IEC 11179, clause 12 describes the basic attributes of metadata items for purposes where a complete metadata registry is not appropriate.

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

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

- a) the definition, specification and contents of metadata registries, including interchanging or referencing among various collections of data elements;
- b) the design and specification of application-oriented data models, databases and message types for data interchange;
- c) the actual use of data in communications and information processing systems;
- d) interchange or reference among various collections of metadata;
- e) the registration and management of semantic artifacts that are useful for data management, data administration, and data analysis;
- f) the interrelation and mapping of concept systems with other concept systems, e.g., to support efforts to converge on consistency through harmonization and vetting activities;
- g) the interrelation of concept systems with data held in relational databases, XML databases, knowledgebases, text, and possibly graph databases deriving from natural language text understanding systems;

- h) the provision of services for semantic computing: Semantics Service Oriented Architecture, Semantic Grids, semantics based workflows, Semantic Web, etc.;
- i) support for addressing semantic web considerations such as AAA (anyone can say anything about anything), non-unique names, and open world assumption;
- j) the capture of semantics with more formal techniques (in addition to natural language) -- First Order Logic (e.g., Common Logic), Description Logics (such as OWL-DL);
- k) support of Application Development and Maintenance;
- l) support of data migration, data mediation;
- m) support of portals, data marts, and data warehouses;
- n) support of data grids and online transaction networks;
- o) ontological reasoning with metadata;
- p) ontology entry point for browsing and searching metadata registries;
- q) capture of associations between the published identifiers used in the ontology(s), and the concepts registered in the registry;
- r) support for Ontology-driven Data Translation;
- s) support for data integration & data interoperation.

Information technology — Metadata registries (MDR) —

Part 3: Registry metamodel and basic attributes

1 Scope

1.1 Scope – Structure of a metadata registry

Clauses 5 through 11 specify the structure of a metadata registry in the form of a conceptual data model.

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

1.2 Scope – Basic attributes of metadata items

Clause 12 specifies basic attributes which are required to describe metadata items, and which might be used in situations where a complete metadata registry is not appropriate (e.g. in the specification of other International Standards).

2 Normative references

The following referenced documents are indispensable for the application 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 11179-6, *Information technology — Metadata registries (MDR) — Part 6: Registration*

3 Terms, definitions and abbreviated terms

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

NOTE 1 An alphabetical list of all terms used in this part of ISO/IEC 11179 is included in Annex A.

NOTE 2 Some definitions listed in this clause have one or more notes; some have a reference to another standard from which the definition is taken; and some have both notes and a reference. Where a definition has both one or more notes and a reference, notes that precede the reference come from the referenced source; notes that follow the reference have been added by this part of ISO/IEC 11179.

3.1 Terms and definitions of metamodel constructs used in this part of ISO/IEC 11179

NOTE This subclause defines the **metamodel constructs** (3.2.81 - units of notation for modelling) used in specifying the registry metamodel in Clauses 6 through 11.

3.1.1

abstract class

⟨metamodel⟩ **class** (3.1.5) that does not provide a complete declaration and typically cannot be instantiated

NOTE 1 An abstract class is intended to be used by other classes as a general class for **specialization** (3.1.16);

NOTE 2 Adapted from ISO/IEC 19505-1:2012, 9.19.1 Classifier.isAbstract.

NOTE 3 Cf. **concrete class** (3.1.8)

3.1.2

association

⟨metamodel⟩ semantic **relationship** (3.1.15) between two **classes** (3.1.5)

NOTE 1 An association is a type of relationship;

NOTE 2 Adapted from ISO/IEC 19505-2:2012, 7.3.3.

3.1.3

association class

⟨metamodel⟩ **association** (3.1.2) that is also a **class** (3.1.5)

NOTE 1 An association class not only connects a set of classes, but also defines a set of features that belong to the association itself;

NOTE 2 Adapted from ISO/IEC 19505-2:2012, 7.3.4.

3.1.4

attribute

⟨metamodel⟩ **characteristic** (3.2.14) of an **object** (3.2.87) or set of objects

3.1.5

class

⟨metamodel⟩ description of a set of **objects** (3.2.87) that share the same **attributes** (3.1.4), operations, methods, **relationships** (3.1.15), and semantics

NOTE Adapted from ISO/IEC 19505-2:2012, 7.3.7.

3.1.6

composite attribute

⟨metamodel⟩ **attribute** (3.1.4) whose **datatype** (3.1.8) is non-atomic

EXAMPLE See *Registration.registration_state* (8.1.5.1.2.1), where *registration_state* is a composite attribute with *Registration_State* (8.1.2.6) as its **composite datatype** (3.1.7).

3.1.7

composite datatype

⟨metamodel⟩ **datatype** (3.1.8) that is also a **class** (3.1.5)

NOTE A composite datatype is used as a datatype for a **composite attribute** (3.1.6).

EXAMPLE See *Registration.registration_state* (8.1.5.1.2.1) where *Registration_State* (8.1.2.6) serves as the composite datatype for the composite attribute *registration_state*.

3.1.8**concrete class**

⟨metamodel⟩ **class** (3.1.5) that can be instantiated

NOTE Cf. **abstract class** (3.1.1).

3.1.9**datatype**

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

[ISO/IEC 11404:2007, 3.12]

3.1.10**generalization**

⟨metamodel⟩ **relationship** (3.1.15) between a more general class (the parent) and a more specific class (the child) that is fully consistent with the general class and that adds additional information

NOTE 1 A generalization is a type of **relationship** (3.1.15);

NOTE 2 The more general class is referred to as the **superclass** (3.1.18);

NOTE 3 The more specific class is referred to as a **subclass** (3.1.17);

NOTE 4 A generalization is directed from the **subclass** to the **superclass**;

NOTE 5 'Fully consistent' means that the **subclass** has all of the **attributes** (3.1.4) and **relationships** (3.1.15) of the **superclass**;

NOTE 6 cf. **specialization** (3.1.16); generalization is the inverse of specialization;

NOTE 7 Adapted from ISO/IEC 19505-2:2012, 7.3.20.

3.1.11**identifier**

⟨metamodel⟩ sequence of characters, capable of uniquely identifying that with which it is associated, within a specified context

NOTE A **name** (3.2.83) should not be used as an identifier because it is not linguistically neutral.

3.1.12**metamodel region**

sub-division of a **package** (3.1.13) used to organize **metadata objects** (3.2.76) for ease of explanation

3.1.13**package**

grouping of **metadata objects** (3.2.76) that provides a **namespace** (3.2.84) for the grouped objects, and allows them to be referenced as a group

NOTE 1 Adapted from ISO/IEC 19505-2:2012, 7.3.37.

3.1.14**primitive datatype**

datatype (3.1.8) that cannot be decomposed into other datatypes without loss of associated semantics

NOTE Adapted from ISO/IEC 11404:2007, 3.44.

3.1.15**relationship**

⟨metamodel⟩ connection among model elements

NOTE 1 In this part of ISO/IEC 11179, a relationship is one of: an **association** (3.1.2), a **generalization** (3.1.10) or a **specialization** (3.1.16);

NOTE 2 Adapted from ISO/IEC 19505-2:2012, 7.3.47.

3.1.16

specialization

<metamodel> **relationship** (3.1.15) between a more general class (the parent) and a more specific class (the child) that is fully consistent with the general class and that adds additional information

NOTE 1 A specialization is a type of **relationship** (3.1.15);

NOTE 2 The more general class is referred to as the **superclass** (3.1.18);

NOTE 3 The more specific class is referred to as a **subclass** (3.1.17);

NOTE 4 A specialization is directed from the **superclass** to the **subclass**;

NOTE 5 'Fully consistent' means that the **subclass** has all of the **attributes** (3.1.4) and **relationships** (3.1.15) of the **superclass**;

NOTE 6 cf. **generalization** (3.1.10); specialization is the inverse of generalization;

NOTE 7 Adapted from ISO/IEC 19505-2:2012, 7.3.20.

3.1.17

subclass

class (3.1.5) that is a **specialization** (3.1.16) of another class, its **superclass** (3.1.18)

NOTE 1 In UML, **subclasses** of a **superclass** are by default incomplete and **disjoint** (3.2.59). This part of ISO/IEC 11179 specifies when subclasses are complete and/or overlapping. Further, when the list of subclasses is intended to be exhaustive, this part of ISO/IEC 11179 shows the superclass as abstract, thus preventing any other subclass being instantiated. An **abstract class** (3.1.1) is indicated by showing the class name in *italics* in any class diagram that uses it.

NOTE 2 A particular class may be a subclass with respect to one relationship, but a superclass with respect to another relationship.

3.1.18

superclass

class (3.1.5) that is a **generalization** (3.1.10) of one or more other classes, its **subclasses** (3.1.17)

NOTE 1 In UML, **subclasses** of a **superclass** are by default incomplete and **disjoint** (3.2.59). This part of ISO/IEC 11179 specifies when subclasses are complete and/or overlapping. Further, when the list of subclasses is intended to be exhaustive, this part of ISO/IEC 11179 shows the superclass as abstract, thus preventing any other subclass being instantiated. An **abstract class** (3.1.1) is indicated by showing the class name in *italics* in any class diagram that uses it.

NOTE 2 A particular class may be a superclass with respect to one relationship, but a subclass with respect to another relationship.

3.2 Terms for concepts used in this part of ISO/IEC 11179

NOTE This subclause provides definitions for terms which are concepts used in the specification of the metadata model in Clauses 5 through 10. Data definitions are included in those clauses. An alphabetical list of terms with links to the corresponding definitions is included in Annex A.

3.2.1

acceptability rating

scale of acceptability

NOTE 1 The following values shall be used: preferred, admitted, deprecated, obsolete and superseded

NOTE 2 Adapted from ISO 10241 :1992, 5.2.2.

3.2.2

administered item

registered item (3.2.105) for which **administrative information** (3.2.3) is recorded

3.2.3

administrative information

<metadata registry> information about the administration of an item in a **metadata registry** (3.2.78)

EXAMPLES creation date, last change date, origin, change description, explanatory comment

3.2.4

arity

number of arguments that a function takes

3.2.5

assertion

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

3.2.6

attached item

registered item (3.2.105) for which **administrative information** (3.2.3) is recorded in another registered item

NOTE This is often a member of a group of registered items that is managed as a whole.

3.2.7

attribute instance

specific instance of an **attribute** (3.1.4)

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

3.2.8

attribute value

value associated with an **attribute instance** (3.2.7)

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

3.2.9

basic attribute

<metadata> **attribute** of a **metadata item** commonly needed in its specification

3.2.10

binary relation

relation (3.2.119) with **arity** (3.2.4) equal to 2 (i.e. whose members all have two ends)

NOTE Most common semantic relations are binary, e.g. *equals*, *less than*, *greater than*, *is part of*, etc. An example of a relation which is not binary is *betweenness*. (e.g. A is between B and C.)

3.2.11

binding

mapping from one framework or specification to another, enabling **data** (3.2.27) and/or commands to be passed between them

3.2.12

boolean

mathematical **datatype** (3.1.8) associated with two-valued logic

[ISO/IEC 11404:2007, 8.1.1 Boolean]

3.2.13

cardinality

number of elements in a set

cf. **multiplicity** (3.2.82)

NOTE Adapted from ISO/IEC 19501:2005, Glossary

3.2.14

characteristic

abstraction of a **property** (3.2.100) of an **object** (3.2.87) or of a set of objects

NOTE Characteristics are used for describing **concepts** (3.2.18).

[ISO 1087-1:2000, 3.2.4]

3.2.15

classifiable item

metadata item (3.2.75) of a type for which classification is supported in a given **metadata registry** (3.2.78)

3.2.16

classification scheme

descriptive information for an arrangement or division of **objects** (3.2.87) into groups based on criteria such as **characteristics** (3.2.14), which the objects have in common

EXAMPLE Origin, composition, structure, application, function, etc.

3.2.17

common attribute

<metadata> **basic attribute** (3.2.9) that is applicable to many or all types of **metadata item** (3.2.75)

3.2.18

concept

unit of knowledge created by a unique combination of **characteristics** (3.2.14)

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

[ISO 1087-1:2000, 3.2.1]

NOTE 2 A concept is independent of its representation.

3.2.19

concept system

set of **concepts** (3.2.18) structured according to the **relations** (3.2.119) among them

[ISO 1087-1:2001, 3.2.11]

3.2.20

conceptual data model

data model (3.2.36) that represents an abstract view of the real world

3.2.21

conceptual domain

CD

concept (3.2.18) that expresses its description or valid instance meanings

3.2.22**conditional**

required under certain specified conditions

NOTE 1 One of three obligation statuses applied to the attributes of metadata items, indicating the conditions under which the attribute is required. See also **mandatory** (3.2.71) and **optional** (3.2.89).

NOTE 2 Obligation statuses apply to metadata items with a **registration status** (3.2.112) of "Recorded" or higher.

3.2.23**contact**

instance of a **role** (3.2.121) of an **individual** (3.2.65) or **organization** (3.2.90) (or **organization part** (3.2.93) or **organization Person** (3.2.95)) to or from whom an information item(s), a material object(s) and/or person(s) can be sent in a specified context

3.2.24**contact information**

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

3.2.25**context**

<designation and definition> setting in which a **designation** (3.2.51) or **definition** (3.2.39) is used

3.2.26**coordinate**

measurement from the origin of a frame of reference

3.2.27**data**

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

NOTE Data can be processed by human or automatic means.

[ISO/IEC 2382-1:1993, 01.01.02]

3.2.28**data element**

<in organization of data> unit of **data** (3.2.27) that is considered in context to be indivisible

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

[ISO/IEC 2382-4:1999, 04.07.01]

NOTE 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).

3.2.29**data element concept**

concept (3.2.18) that is an **association** (3.1.2) of a **property** (3.2.100) with an **object class** (3.2.88)

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

NOTE 2 A data element concept may also be associated with zero or more **conceptual domains** (3.2.21) each of which expresses its **value meanings** (3.2.141).

NOTE 3 A data element concept may also be associated with zero or more **data elements** (3.2.28) each of which provide representation for the data element concept via its associated **value domain** (3.2.140).

3.2.30

data element concept property

property (3.2.100) of a **data element concept** (3.2.29)

3.2.31

data element concept domain

conceptual domain (3.2.21) of a **data element concept** (3.2.29)

3.2.32

data element concept object class

object class (3.2.88) of a **data element concept** (3.2.29)

3.2.33

data element derivation

application of a **derivation rule** (3.2.45) to one or more input **data elements** (3.2.28) to derive one or more output data elements

3.2.34

data element example

representative illustration of a **data element** (3.2.28)

3.2.35

data element precision

degree of specificity for a **data element** (3.2.28)

NOTE Expressed as a number of decimal places to be used in any associated data element values.

3.2.36

data model

graphical and/or lexical representation of **data** (3.2.27), specifying their properties, structure and inter-relationships

3.2.37

date

datatype (3.1.8) whose values are points in time to the resolution: year, month, day

NOTE 1 Adapted from ISO/IEC 11404:2007, 8.1.6 date and time;

NOTE 2 See also 6.2.3 Date datatype.

3.2.38

datetime

datatype (3.1.8) whose values are points in time to the resolution: year, month, day, hour, minute, second, and optionally fractions thereof

NOTE 1 Adapted from ISO/IEC 11404:2007, 8.1.6 date and time;

NOTE 2 See also 6.2.4 Datetime datatype.

3.2.39

definition

representation of a **concept** (3.2.18) by a descriptive statement which serves to differentiate it from related concepts

[ISO 1087-1:2000, 3.3.1]

3.2.40**definition**

<designatable item> representation of a **designatable item** (3.2.50) by a descriptive statement which, in a given **language** (3.2.68) and **context(s)** (3.2.25) serves to differentiate it from related designatable items

NOTE See also **definition** (3.2.39 above).

3.2.41**definition context**

<designatable item> **context** (3.2.25) in which the **definition** (3.2.39) is applicable

3.2.42**definition language**

language (3.2.68) used to write the **definition text** (3.2.44)

3.2.43**definition source reference**

reference to the source from which the **definition** (3.2.39) is taken

3.2.44**definition text**

text of the **definition** (3.2.39)

3.2.45**derivation rule**

logical, mathematical, and/or other operations specifying derivation

3.2.46**derivation rule notation**

notation (3.2.86) used to specify the **derivation rule** (3.2.45)

3.2.47**derivation rule specification**

text of a specification of **data element derivation** (3.2.33)

3.2.48**described conceptual domain**

conceptual domain (3.2.21) that is specified by a description or specification, such as a rule, a procedure, or a range (i.e. interval)

3.2.49**described value domain**

value domain (3.2.140) that is specified by a description or specification, such as a rule, a procedure, or a range (i.e. interval)

3.2.50**designatable item**

identified item (3.2.64) which can have **designations** (3.2.51) and/or **definitions** (3.2.40)

3.2.51**designation**

representation of a **concept** (3.2.18) by a **sign** (3.2.123) which denotes it

[ISO 1087-1:2000, 3.4.1]

NOTE See also 3.2.52 designation <designatable item>

3.2.52

designation

<designatable item> representation of a **designatable item** (3.2.50) by a **sign** (3.2.123) which denotes it

NOTE This contextualized designation is as specified in **designation** (3.2.51) or a **name** (3.2.83).

3.2.53

designation acceptability

rating of the acceptability of the **designation** (3.2.51) in the specified **context** (3.2.25)

3.2.54

designation context

<designatable item> **context** (3.2.25) in which a **designation** (3.2.51) is applicable

3.2.55

designation language

<designatable item> **language** (3.2.68) or dialect in which a **sign** (3.2.123), usually a **name** (3.2.83), is expressed

3.2.56

designation namespace

namespace (3.2.84) to which a **designation** (3.2.51) is bound

3.2.57

designation sign

<designatable item> **sign** (3.2.123) of the **designation** (3.2.51)

3.2.58

dimensionality

set of equivalent **units of measure** (3.2.138)

NOTE 1 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 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:

- 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 $^{\circ}\text{F} = (9/5) * (^{\circ}\text{C}) + 32$.
- the same operations can be performed on both values.

NOTE 3 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, 20K is twice as warm as 10K, but 20 $^{\circ}\text{C}$ is not twice as warm as 10 $^{\circ}\text{C}$.

NOTE 4 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 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 31-0 calls these "quantities of the same kind".

NOTE 6 ISO 31-0 specifies physical dimensions (e.g. length, mass, velocity). This part of ISO/IEC 11179 also permits non-physical dimensions (e.g. value dimensions such as: currency, quality indicator). The present concept of dimensionality equates to what ISO 31 calls Dimensional Product, rather than to Dimension.

3.2.59**disjoint**

<set theory> having no elements in common

3.2.60**enumerated conceptual domain**

conceptual domain (3.2.21) that is specified by a list of all its **value meanings** (3.2.141)

NOTE No ordering of the value meanings is implied.

3.2.61**enumerated value domain**

value domain (3.2.140) that is specified by a list of all its **permissible values** (3.2.96)

NOTE No ordering of the permissible values is implied.

3.2.62**extension**

<11179-3> feature not defined by this part of ISO/IEC 11179

<registry metamodel> **class** (3.1.5), **attribute** (3.1.4) or **relationship** (3.1.15) that an implementation of a **metadata registry** (3.2.78) provides that is not defined by this part of ISO/IEC 11179

NOTE This part of ISO/IEC 11179 specifies **slots** (3.2.124) as a mechanism for extending **identified items** (3.2.64).

3.2.63**identification scheme**

system for allocating **identifiers** (3.1.11) to registered **objects** (3.2.87)

[ISO/IEC 6523-1:1998, 3.6]

3.2.64**identified item**

metadata item (3.2.75) identified in a **metadata registry** (3.2.78)

3.2.65**individual**

single human being

3.2.66**integer**

mathematical **datatype** (3.1.8) comprising the exact integral values

[ISO/IEC 11404:2007, 8.1.7]

3.2.67**international code designator**

identifier (3.1.11) of an **organization identification scheme** (3.2.91)

NOTE 1 Adapted from ISO/IEC 6523-1:1998, 3.8.

NOTE 2 See also ISO/IEC 11179-6.

3.2.68**language**

system of **signs** (3.2.123) for communication, usually consisting of a vocabulary and rules

[ISO 5127:2001, 1.1.2.01]

3.2.69

link

member of a **relation** (3.2.119)

3.2.70

link end

end of a **link** (3.2.69), identifying the **relation role** (3.2.119) played by a **concept** (3.2.18) in the link

3.2.71

mandatory

always required

NOTE 1 One of three obligation statuses applied to the attributes of metadata items, indicating the conditions under which the attribute is required; see also **conditional** (3.2.22) and **optional** (3.2.89);

NOTE 2 Obligation statuses apply to **metadata items** (3.2.75) with a **registration status** (3.2.112) of "recorded" or higher.

3.2.72

measure class

set of equivalent **units of measure** (3.2.138) for association with one or more **dimensionalities** (3.2.58)

3.2.73

meronomy

type of hierarchy which deals with part-whole relationships

NOTE cf. **taxonomy** (3.2.135)

3.2.74

metadata

data (3.2.27) that defines and describes other data

3.2.75

metadata item

instance of a **metadata object** (3.2.76)

NOTE 1 In all parts of ISO/IEC 11179, this term is applied only to instances of metadata objects described by the metamodel in Clauses 5 through 11 of this part of ISO/IEC 11179. Examples include instances of *Data_Elements* (11.5.2.1), *Data_Element_Concepts* (11.2.2.3), *Permissible_Values* (11.3.2.7) etc.

NOTE 2 A metadata item has associated **attributes** (3.1.4), as appropriate for the metadata object it instantiates.

3.2.76

metadata object

object type defined by a **metamodel** (3.2.80)

NOTE In all parts of ISO/IEC 11179, this term is applied only to metadata objects described by the metamodel in Clauses 5 through 11 of this part of ISO/IEC 11179. Examples include *Data_Elements* (11.5.2.1), *Data_Element_Concepts* (11.2.2.3), *Permissible_Values* (11.3.2.7) etc.

3.2.77

metadata register

information store or database maintained by a **metadata registry** (3.2.78)

3.2.78

metadata registry

MDR

information system for registering **metadata** (3.2.74)

NOTE The associated information store or database is known as a **metadata register** (3.2.77).

3.2.79**metadata registry product**

particular information system for implementing a **metadata registry** (3.2.78)

3.2.80**metamodel**

model that specifies one or more other models

3.2.81**metamodel construct**

unit of **notation** (3.2.86) for modelling

NOTE The metamodel constructs used in this part of ISO/IEC 11179 are defined in 3.1.

3.2.82**multiplicity**

specification of the range of allowable **cardinalities** (3.2.13) that a set may assume

NOTE 1 Multiplicity specifications may be given for roles within **associations** (3.1.2)

NOTE 2 A multiplicity is a (possibly infinite) subset of the nonnegative integers

NOTE 3 Adapted from ISO/IEC 19501:2005, Glossary

3.2.83**name**

designation (3.2.51) of an **object** (3.2.87) by a linguistic expression

[ISO 1087:1990, 5.3.1.3 and ISO/IEC 15944-1:2002, 3.35]

3.2.84**namespace**

set of **designations** (3.2.51) and/or **scoped identifiers** (3.2.122) for a particular business need

NOTE The term **namespace** is used in this International Standard because it is in common use, even though the concept is being applied to identifiers as well as names.

3.2.85**naming convention**

specification of how **signs** (3.2.123) of **designations** (3.2.51) and/or **scoped identifiers** (3.2.122) are formulated

NOTE A naming convention can apply to scoped identifiers when they are included in the associated **namespace** (3.2.84)

3.2.86**notation**

formal syntax and associated semantics

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

NOTE Formal syntax is often intended for machine processing.

3.2.87**object**

anything perceivable or conceivable

NOTE 1 Objects may also be material (e.g. an engine, a sheet of paper, a diamond), immaterial (e.g. a conversion ratio, a project plan) or imagined (e.g. a unicorn);

NOTE 2 Adapted from ISO 1087-1:2000, 3.1.1.

3.2.88

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

3.2.89

optional

permitted but not required

NOTE 1 One of three obligation statuses applied to the **attributes** (3.1.4) of **metadata items** (3.2.75), indicating the conditions under which the attribute is required. See also **conditional** (3.2.22) and **mandatory** (3.2.71).

NOTE 2 Obligation statuses apply to **metadata items** (3.2.75) with a **registration status** (3.2.112) of "recorded" or higher.

3.2.90

organization

unique framework of authority within which **individuals** (3.2.65) act, or are designated to act, towards some purpose

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

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

NOTE 2 Adapted from ISO/IEC 6523-1:1998, 3.1

3.2.91

organization identification scheme

identification scheme (3.2.63) dedicated to the unique identification of **organizations** (3.2.90)

[ISO/IEC 6523-1:1998, 3.7]

3.2.92

organization identifier

identifier (3.1.11) assigned to an **organization** (3.2.90) within an **organization identification scheme** (3.2.91), and unique within that scheme

[ISO/IEC 6523-1:1998, 3.10]

3.2.93

organization part

any department, service or other entity within an **organization** (3.2.90) which needs to be identified for information exchange

[ISO/IEC 6523-1:1998, 3.2]

3.2.94**organization part identifier****opi**

identifier (3.1.11) allocated to a particular **organization part** (3.2.93)

NOTE See also ISO/IEC 11179-6.

[ISO/IEC 6523-1:1998, 3.11]

3.2.95**organization Person**

organization part (3.2.93) which has the properties of a **Person** (3.2.97) and thus is able to make commitments on behalf of that **organization** (3.2.90)

NOTE 1 An **organization** can have one or more **organization Persons**.

NOTE 2 An **organization Person** is deemed to represent and act on behalf of the **organization** and to do so in a specified capacity.

NOTE 3 An **organization Person** can be a "natural person" such as an employee or officer of the **organization**.

NOTE 4 An **organization Person** can be a "legal person", i.e., another **organization**.

[ISO/IEC 15944-1:2002, 3.46]

3.2.96**permissible value**

designation (3.2.51) of a **value meaning** (3.2.141)

NOTE A permissible value may be associated with one or more **enumerated value domains** (3.2.61).

3.2.97**Person**

entity, i.e., a natural or legal person, recognized by law as having legal rights and duties, able to make commitment(s), assume and fulfil resulting obligation(s), and able of being held accountable for its action(s)

NOTE 1 Synonyms for "legal person" include "artificial person", "body corporate", etc., depending on the terminology used in competent jurisdictions.

NOTE 2 Person is capitalized to indicate that it is being utilized as formally defined in the standards and to differentiate it from its day-to-day use.

NOTE 3 Minimum and common external constraints applicable to a business transaction often require one to differentiate among three common subtypes of Person, namely "individual", "organization", and "public administration"

[ISO/IEC 15944-1:2002, 3.47]

3.2.98**phone number**

telephone number

NOTE Specified by ITU-T Recommendation E.164 (2005-02), the international public telecommunications numbering plan.

3.2.99**postal address**

set of information which, for a postal item, allows the unambiguous determination of an actual or potential delivery point, usually combined with the specification of an addressee and/or a mailer

[UPU S42]

3.2.100

property

quality common to all members of an **object class** (3.2.88)

3.2.101

quantity

value with an associated **unit of measure** (3.2.138)

NOTE 32° Fahrenheit and 0° Celsius are quantities, and they are equivalent values in different measuring systems.

3.2.102

reference document

document that provides pertinent details for consultation about a subject

3.2.103

reflexivity

characterization of a **binary relation** (3.2.10) as: reflexive, irreflexive or antireflexive

NOTE 1 A binary relation, R, is reflexive if for all x, R(x,x) is true. Equality is an example of a reflexive relation.

NOTE 2 A binary relation, R, is irreflexive if it is not reflexive. i.e., R(x,x) is not necessarily true for all x.

NOTE 3 A binary relation, R, is antireflexive if for all x, R(x,x) is false. Inequality is an example of an antireflexive relation. An antireflexive relation is also irreflexive, but antireflexive is a more specific characterization.

3.2.104

register

information store or database maintained by a **registry** (3.2.113)

3.2.105

registered item

metadata item (3.2.75) that is recorded and managed in a **metadata registry** (3.2.78)

3.2.106

registrar

representative of a **registration authority** (3.2.109)

3.2.107

registrar identifier

identifier (3.1.11) for the **registrar** (3.2.106)

3.2.108

registration

<generic>inclusion of an item in a **registry** (3.2.113)

<metadata registry> inclusion of a **metadata item** (3.2.75) in a **metadata registry** (3.2.78)

3.2.109

registration authority

RA

organization (3.2.90) responsible for maintaining a **register** (3.2.104)

3.2.110

registration authority identifier

identifier (3.1.11) assigned to a **registration authority** (3.2.109)

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

3.2.111**registration state**

<registration> information about the **registration** (3.2.108) of an **administered item** (3.2.2)

3.2.112**registration status**

designation (3.2.51) of the status in the **registration** (3.2.108) life-cycle of an **administered item** (3.2.2)

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

3.2.113**registry**

information system for **registration** (3.2.108)

3.2.114**registry instance**

<generic> implementation of a **registry product** (3.2.117) and instance of a **registry** (3.2.113)

<metadata registry> implementation of a **metadata registry product** (3.2.79) and instance of a **metadata registry** (3.2.78)

3.2.115**registry item**

<generic> item recorded in a **registry** (3.2.113)

<metadata registry> **metadata item** (3.2.75) recorded in a **metadata registry** (3.2.78)

3.2.116**registry metamodel**

metamodel (3.2.80) specifying the model for a **registry** (3.2.113)

3.2.117**registry product**

particular information system for implementing a **registry** (3.2.113)

3.2.118**related metadata reference**

reference from one **metadata item** (3.2.75) to another

NOTE A **registration authority** (3.2.109) could choose to use a *Reference_Document* (8.1.3.3), an *administrative_note* (8.1.2.6.2.4) or an *explanatory_comment* (8.1.2.2.3.4) to record a related metadata reference.

3.2.119**relation**

sense in which **concepts** (3.2.18) may be connected, via constituent roles.

EXAMPLE causality is a relation with two constituent roles: cause and effect.

NOTE The related concepts may be general or individual concepts

3.2.120**relation role**

role that a **concept** (3.2.18) plays in a **relation** (3.2.119)

3.2.121**role**

specified responsibilities

3.2.122

scoped identifier

identifier (3.1.11) of an **identified item** (3.2.64) within a specified **namespace** (3.2.84)

NOTE A **namespace** provides the scope within which the **scoped identifier** uniquely identifies the **identified item**.

3.2.123

sign (noun)

textual string or symbol that can be used to denote a **concept** (3.2.18)

3.2.124

slot

container for an **extension** (3.2.62) to an **identified item** (3.2.64)

3.2.125

stewardship

<metadata> responsibility for the maintenance of **administrative information** (3.2.3) applicable to one or more **administered items** (3.2.2)

NOTE The responsibility for the **registration** (3.2.108) of **metadata** (3.2.74) may be different from the responsibility for **stewardship** of **metadata**.

3.2.126

stewardship contact

contact (3.2.23) information associated with a **stewardship** (3.2.125)

3.2.127

stewardship organization

organization (3.2.90) that maintains **stewardship** (3.2.125) of an **administered item** (3.2.2)

3.2.128

stewardship record

record of a **stewardship organization** (3.2.127) and a **stewardship contact** (3.2.126) involved in the **stewardship** (3.2.125) of an **administered item** (3.2.2)

3.2.129

string

family of **datatypes** (3.1.8) which represent strings of symbols from standard character-sets

[ISO/IEC 11404:2007 10.1.5 Character String]

NOTE The syntax and semantics of the String datatype (6.2.11) are as defined in ISO/IEC 11404:2007 10.1.5 Character String

3.2.130

submission

act of submitting a **metadata item** (3.2.75) for registration in a **metadata registry** (3.2.78)

3.2.131

submission contact

contact (3.2.23) information associated with a **submission** (3.2.130)

3.2.132

submission organization

organization (3.2.90) that submits a **metadata item** (3.2.75) for **registration** (3.2.108)

3.2.133

submission record

record of a **submission organization** (3.2.132) and a **submission contact** (3.2.131) involved in the **submission** (3.2.130) of a **metadata item** (3.2.75) for **registration** (3.2.108)

3.2.134**symmetry**

characterization of a **binary relation** (3.2.10) as: symmetric, asymmetric or antisymmetric

NOTE 1 A binary relation, R , is symmetric if for all x, y : $R(x,y)$ implies $R(y,x)$.

EXAMPLE Symmetric relations include: 'equals', 'not equals', 'within-2-miles-of', etc.

NOTE 2 Symmetry does not imply **reflexivity** (3.2.103).

EXAMPLE the 'inequality' relation is symmetric, but antireflexive.

NOTE 3 A binary relation, R , is asymmetric if for all x, y : $R(x,y)$ does not imply $R(y,x)$.

EXAMPLE Asymmetric relations include: less than, likes, father of, etc.

NOTE 4 A binary relation, R , is anti-symmetric if for all x, y : $R(x,y)$ implies not $R(y,x)$. 'An antisymmetric relation is also asymmetric, but antisymmetric is a more specific characterization.

EXAMPLE 'less than' is an antisymmetric relation.

NOTE 5 An asymmetric relation is not necessarily antisymmetric

EXAMPLE Less than or equals.

3.2.135**taxonomy**

type of hierarchy which deals with generalization/specialization relationships.

NOTE cf. **meronomy** (3.2.73)

3.2.136**text**

paragraph, page or document that can be used to define or describe an **object** (3.2.87)

NOTE 1 Text is **data** (3.2.27) in the form of characters, symbols, words, phrases, paragraphs, sentences, tables, or other character arrangements, intended to convey a meaning, and whose interpretation is essentially based upon the reader's knowledge of some natural language or artificial language

NOTE 2 Adapted from ISO/IEC 2382-23:1994

NOTE 3 See also the Text datatype (6.2.12).

3.2.137**transitive relation**

binary relation (3.2.10) R over a set X is transitive if whenever an element a is related to an element b , and b is in turn related to an element c , then a is also related to c .

3.2.138**unit of measure**

(value domain) actual units in which the associated values are measured

NOTE 1 ISO 31-0:1982 specifies a system of physical measurement (the International System of Units, SI). Physical measurement is only one type of measurement. Value measurement is another type of measurement. This part of ISO/IEC 11179 permits the use of any appropriate system of measurement.

NOTE 2 The **dimensionality** (3.2.58) of the associated **conceptual domain** (3.2.21) must be appropriate for the specified **unit of measure**.

3.2.139

unit of measure dimensionality

dimensionality (3.2.58) that specifies the equivalence relation that applies to all values representing a particular unit

3.2.140

value domain

VD

set of **permissible values** (3.2.96)

NOTE 1 The **value domain** provides representation, but has no implication as to what **data element concept** (3.2.29) the values might be associated with nor what the values mean

NOTE 2 The **permissible values** may either be enumerated or expressed via a description.

3.2.141

value meaning

semantic content of a possible value

NOTE The representation of value meanings in a **registry** (3.2.113) shall be independent of (and shall not constrain) their representation in any corresponding **value domain** (3.2.140).

3.2.142

version

unique version **identifier** (3.1.11) of the **scoped identifier** (3.2.122)

3.3 Abbreviated terms

3.3.1

CD

Conceptual Domain

3.3.2

CL

Common Logic

3.3.3

CLIF

Common Logic Interchange Format

3.3.4

DE

Data Element

3.3.5

DEC

Data Element Concept

3.3.6

MDR

metadata registry

3.3.7

mece

mutually exclusive and collectively exhaustive

3.3.8

opi

organization part identifier

3.3.9

ORM

Object Role Modelling

3.3.10

OWL

Web Ontology Language

3.3.11

OWL-DL

OWL Description Logic

3.3.12

RA

Registration Authority

3.3.13

RDF

Resource Description Framework

3.3.14

SBVR

Semantics of Business Vocabulary and Business Rules

3.3.15

SKOS

Simple Knowledge Organization System

3.3.16
Turtle
 Terse RDF Triple Language

3.3.17
UML
 Unified Modeling Language

3.3.18
UPU
 Universal Postal Union

3.3.19
URI
 Universal Resource Identifier

3.3.20
VD
 Value Domain

3.3.21
W3C
 World Wide Web Consortium

3.3.22
XCL
 eXtended Common Logic markup language

3.3.23
XML
 eXtensible Markup Language

3.3.24
XTML
 eXtensible Topic Maps

4 Conformance

4.1 Overview of conformance

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

This part of ISO/IEC 11179 also prescribes a list of basic attributes (see clause 12) for situations where a full conceptual model is not required or not appropriate.

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

Conformance statements with respect to this standard must also be explicit as to which portions of this standard conformity is being claimed. This may be done in some cases simply by reference to one or more of the clauses. In other cases, conformance may instead be claimed to one or more of the standard profiles (see Annex H), which specify combinations of multiple clauses, and how they are to be fitted together.

When a **registry product** (3.2.117) makes a conformance claim, the product must support all the associated functionality, and must enable the enforcement of the associated constraints. When a **registry instance** (3.2.114) makes a conformance claim, it must actually enforce the associated constraints.

4.2 Degree of conformance

4.2.1 General

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

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

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

4.2.2 Strictly conforming implementations

A strictly conforming implementation:

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

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

4.2.3 Conforming implementations

A conforming implementation:

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

NOTE 1 All strictly conforming implementations are also conforming implementations.

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

4.3 Conformance by clause

Conformance claims may be limited to individual clauses 6-12 of this standard. Clauses 7-11 are all dependent upon one or more other clauses of this standard (see Figure 1 on p. 26), so conformance to any of these clauses must be understood to imply conformance also to relevant provisions specified in one or more preceding clauses.

Conformance may be claimed for a set of data structures and/or datatypes for clauses:

- clause 7 Identification, Designation and Definition
- clause 9 Concepts
- clause 10 Binary Relations
- clause 11 Data Description .

Conformance may be claimed for a register or registry (a set of administered metadata), or for a registry software system for clauses:

- clause 8 Registration
- clause 12 Basic attributes.

4.4 Registry conformance

4.4.1 Overview

Registries shall claim conformance to either clause 8 or clause 12. Registries conforming to clause 8 may additionally claim conformance to one of the standard profiles specified in clause Annex H, each of which incorporates some subset of the clauses 7 and 9-11.

4.4.2 Standard profiles for edition 3 registries

Implementers of either a generic registry platform (software system) which is customizable for a range of registered content types, or of a registry or registry software system supporting some specific range of registered content types not specified in this standard, might simply claim conformance to clause 8. Alternatively, the following standard profiles are defined, specifying how additional clauses should be combined with clause 8:

- **Concept Systems Registry:** Implements clauses 7-9, and also satisfies the following additional provisions:
 - All Concepts and Concept_Systems must be Registered_Items.
 - All Registered_Items must also be Designatable_Items and Classifiable_Items.
- **Extended Concept Systems Registry:** Implements clause 10, in addition to all provisions of the Concept Systems Registry profile.
- **Metadata Registry:** Implements clause 11, in addition to all provisions of the Concept Systems Registry profile, and also satisfies the following additional provision:
 - All Data_Elements, Value_Domains, and Derivation_Rules must be Registered_Items.
- **Extended Metadata Registry:** Implements all of clauses 7-11, and also satisfies all provisions of the Metadata Registry profile.

4.5 Obligation

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

For the purpose of conformance:

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

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

4.6 Implementation conformance statement (ICS)

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

- a) whether it conforms or strictly conforms;
- b) which clauses are supported;
- c) which registry type is supported
- d) what extensions, if any, are supported or used.

4.7 Roles and responsibilities for registration

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

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

5 Structure of a metadata registry

5.1 Metamodel for a metadata registry

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

This part of ISO/IEC 11179 uses a metamodel to describe the 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.

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

The structure described by this metamodel may be distributed over several implementations. These implementations might be databases, data repositories, metadata registers, metadata registries, dictionaries, etc. No particular technology is implied. Implementations may utilize technologies including, but not limited to: relational database, XML database, object oriented systems, or RDF/OWL.

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

5.2 Application of the metamodel

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

- to provide a unified view of concepts, terms, value domains and value meanings;
- to promote a common understanding of the data described;
- to provide the specification at a conceptual level to facilitate the sharing and reuse of the contents of implementations.

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

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

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

5.3 Specification of the metamodel

5.3.1 Terminology used in specifying the metamodel

When using a model to specify another model, it is easy for the reader to become confused about which model is being referred to at any particular point. To minimize this confusion, this part of ISO/IEC 11179 deliberately uses different terms in the model being specified from those used to do the specification.

The registry metamodel is specified using a subset of the Unified Modelling Language (UML) Version 2.4.1. This part of ISO/IEC 11179 uses the term **metamodel construct** (3.2.81) for the UML model constructs it uses, but **metadata object** (3.2.76) for the model constructs it specifies. The metamodel constructs used are: classes, associations, association classes, attributes, composite attributes and composite datatypes, and these are defined in 3.1. The specified metadata objects are defined in clauses 6 through 11. The concepts that the metadata objects represent are defined in clause 3.2.

However, there are certain parallels between the two metamodels. For example, the *Object_Class* (11.2.2.1) specified in the registry metamodel is similar to the metamodel construct **class** (3.1.5), and the *Property* (11.2.2.2) specified in the registry metamodel is similar to the metamodel construct **attribute** (3.1.4). The different terms are used to make it clear which model is being referred to, not because they represent different concepts. One term that this part of ISO/IEC 11179 uses at both levels is “datatype”, but the level to which it applies should be apparent from the context in which it is used.

5.3.2 Choice of fonts

In clauses 6 through 11 this specification uses:

- **bold** font to highlight terms which represent concepts defined in clause 3;
- *italic* font to highlight terms which represent metadata objects specified by the metamodel.

EXAMPLE *Concept* (9.1.2.1) is a class each instance of which models a **concept** (3.2.18).

5.3.3 Use of UML Packages

For descriptive and conformance purposes, the metamodel is organized into packages:

- Basic package (clause 6) – contains simple classes that are reused by other packages
- Identification, designation and definition package (clause 7) – contains classes that enable the contents of a registry to be identified, named or otherwise designated, and defined. This package is sub-divided into two regions:
 - Identification metamodel region (see 7.2)

- Designation and Definition metamodel region (see 7.3)
- Registration package (clause 8) – contains classes that enable metadata items to be registered. These are specified as the Registration metamodel region (see 8.1).
- Concepts package (clause 9) – contains classes that enable concepts to be related. This package is subdivided into two regions:
 - Concepts metamodel region (see 9.1)
 - Classification metamodel region (see 9.2)
- Binary Relations package (clause 10) – contains a specialization of the Relations class from the Concepts package, specifically for binary relations. The separation is for conformance purposes.
- Data Descriptions package (clause 11) – contains classes that enable the description of specific metadata objects:
 - Data Element Concept metamodel region (see 11.2)
 - Conceptual and Value_Domain metamodel region (see 11.3)
 - Measurement metamodel region (see 11.4)
 - Data_Element metamodel region (see 11.5).

5.3.4 Package dependencies

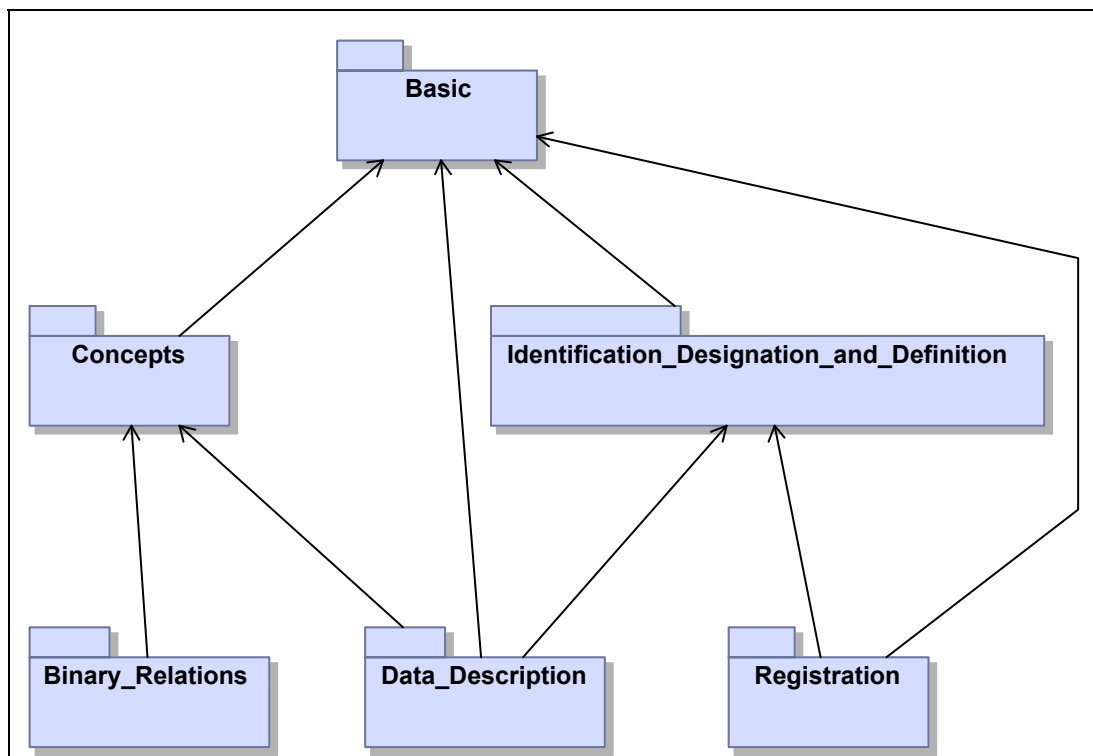


Figure 1 — Package dependencies

Figure 1 illustrates the dependencies among the packages. The lines in the figure illustrate dependencies in the direction of the arrow. In order to implement a package that has dependencies, the packages on which it is dependent must also be implemented. The dependencies are of three types:

- a) Subclassing from classes in another package,
e.g. *Conceptual_Domain* (11.2.2.4), *Data_Element_Concept* (11.2.2.3), *Object_Class* (11.2.2.1) and *Property* (11.2.2.2) in the Data Element Concept region (11.2) of the Data_Description package (11) are all subclassed from the *Concept* class (9.1.2.1) in the Concepts package (9).
- b) Relationship between classes,
e.g. *Registered_Item* (8.1.2.1) in the Registration package (8) has a relationship with *Reference_Document* (6.3.7) in the Basic package (6).
- c) Some attributes use a predefined datatype or a class from another package as a datatype
e.g. the *contact* attribute (8.1.2.7.2.2) of the *Stewardship_Record* class (8.1.2.7) in the Registration package (8) uses the *Contact* class (6.3.2) of the Basic package (6) as a datatype.

Conformance options are specified in clause 4 and standard conformance profiles in Annex H.

5.3.5 Use of UML Class diagrams and textual description

This standard uses both text and UML class diagrams 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 such time as 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 such time as a correction is made to make them consistent. For associations, the description of the association itself takes precedence over any description in the associated classes.

A consolidated UML class hierarchy is included as Annex B.

5.4 Types, instances and values

When considering data and metadata, it is important to distinguish between types of data/metadata, and instances of these types and their associated values. The metamodel specifies types of classes, attributes and associations. Any particular instance of one of these will be of a specific type, and at any point in time, that instance will have a specific value (possibly null). As examples, this document defines *attribute instance* and *attribute value*, but the same principle applies to classes, relationships and all other metamodel constructs defined in 3.1.

NOTE In UML, **subclasses** (3.1.17) of a **superclass** (3.1.18) are by default not **disjoint** (3.2.59). This standard specifies when subclasses are required to be disjoint. Further, when the list of subclasses is intended to be exhaustive, this standard shows the superclass as abstract, thus preventing any other subclass being instantiated. An abstract class is indicated by showing the class name in *italics* in any class diagram that uses it.

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*. A *metadata registry* will be populated with instances of these *metadata objects* (referred to as *metadata items*), which in turn define types of data, e.g. in an application database. In other words, instances of metadata specify types of application level data. In turn, the application database will be populated by the real world data as instances of those defined metadata object types.

NOTE ISO/IEC TR 10032:2003 Reference model for data management explains the concepts of different levels of modelling, as does ISO/IEC 10027:1990 IRDS Framework.

5.5 Types of items in an ISO/IEC 11179 metadata registry

5.5.1 Overview of types of items

Figure 2 shows the types of items specified by this Part of this International Standard. These types are explained in subsequent clauses. In the Figure, the notation <<type>> indicates the use of the <<type>> stereotype as specified in ISO/IEC 19505-2:2012 OMG UML Part 2: Superstructure Annex C.1 Standard Profile L2.

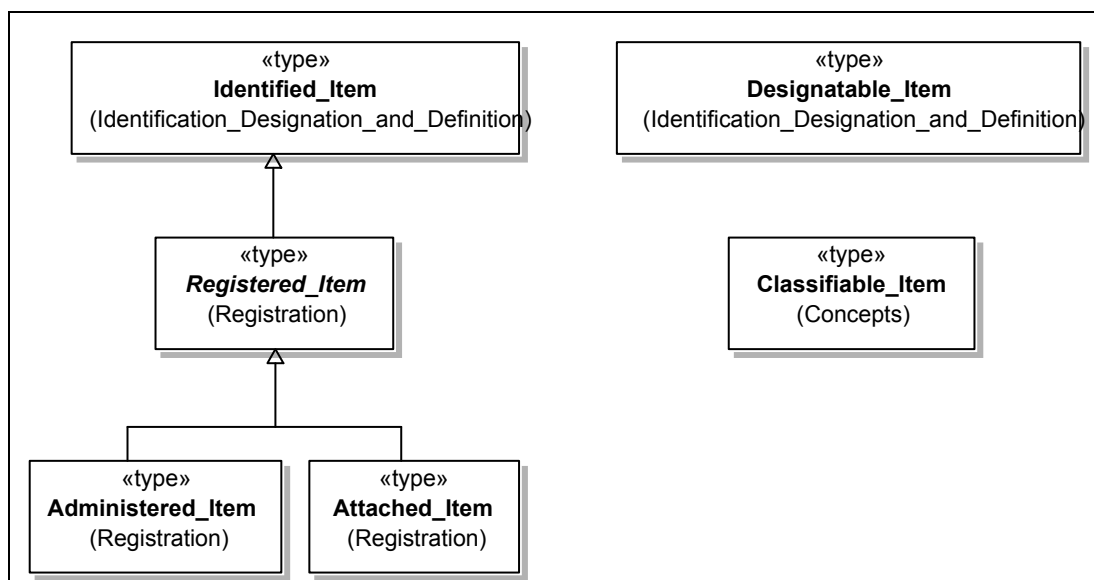


Figure 2 — Types of items

Any **metadata item** (3.2.75) entered into a **metadata registry** (3.2.78) may be extended by one or more of the above types, as follows:

- Any metadata item that is to be retrieved directly (as opposed to indirectly through a related item), shall be an *Identified_Item* (see 7.2.2.1), so the item can be referenced. An example of metadata items that might not be explicitly identified are the *Permissible Values* (11.3.2.7) within a *Value Domain* (11.3.2.5).
- Any metadata item that is to be designated (named) and/or defined shall be a *Designatable_Item* (7.3.2.2).

NOTE The separation of Designation and Definition from Identification has been done to better harmonize with ISO/IEC 19763-2, in which ModelElements are identified and are administered, but are not (required to be) designated or defined.

- Any metadata item that is to be registered in the registry shall be a *Registered_Item* (8.1.2.1). *Registered_Item* is an abstract class, which means that each such item must be instantiated as one of the subclasses: *Administered_Item* (8.1.2.2), or *Attached_Item* (8.1.2.3). These subclasses are mutually exclusive and collectively exhaustive (mece).
- Any metadata item that is to be classified in a **classification scheme** (3.2.16) shall be a *Classifiable_Item* (see 9.2.2.1).

A *Registration_Authority* (8.1.2.5) responsible for the registry shall determine which metadata items should become *Identified_Items* (7.2.2.1), *Registered_Items* (8.1.2.1), *Designatable_Items* (7.3.2.2) and/or *Classifiable_Items* (9.2.2.1), within the constraints of any conformance claim (see clause 4.) that is made for the registry.

NOTE The precise mechanism by which metadata items are extended by the above types is implementation-defined.

Any metadata item that has been extended by one or more of the above types shall inherit the relationships of the type(s), which can thus be navigated from the item (e.g. the *identification* (7.2.3.1) association from *Identified_Item* (7.2.2.1).

5.5.2 Rules for types of items

Table 1 — Rules for Types of Items

Rule #	Rule Description
1	A metadata item is <u>identified</u> if it has one or more <i>identification</i> (7.2.3.1) associations, each with a <i>Scoped_Identifier</i> (7.2.2.2) class.
2	A metadata item is <u>designated</u> if it has one or more <i>item_designation</i> (7.3.4.4) associations, each with a <i>Designation</i> (7.3.2.3) class.
3	A metadata item is <u>defined</u> if it has one or more <i>item_definition</i> (7.3.4.3) associations, each with a <i>Definition</i> (7.3.2.4) class.
4	A metadata item is <u>classified</u> if it has one or more <i>Classification</i> (9.2.3.1) association classes, each with a <i>Concept</i> (9.1.2.1) class in a <i>Concept_System</i> (9.1.2.2).
5	A metadata item is <u>registered</u> if it has one or more <i>submission</i> (8.1.6.5) associations, each with a <i>Submission_Record</i> (8.1.2.8) class. The metadata item must also be <u>identified</u> as described by rule 1 and either <u>administered</u> as described by rule 6, or <u>attached</u> as described by rule 7.
6	A metadata item is <u>administered</u> if it has exactly one <i>stewardship</i> (8.1.6.4) association with a <i>Stewardship_Record</i> (8.1.2.7) class, and one or more <i>Registration</i> (8.1.5.1) association classes, each with a <i>Registration_Authority</i> (8.1.2.5) class. The metadata item must also be <u>registered</u> as described by rule 5. The metadata item cannot also be <u>attached</u> as described by rule 7.
7	A metadata item is <u>attached</u> if it has exactly one <i>attachment</i> (8.1.6.1) association with an <i>Administered_Item</i> (8.1.2.2) class. The metadata item cannot also be <u>administered</u> as described by rule 6.

Table 2 on p. 30 is a compressed decision table that shows the rules that apply to the various types of item.

The upper section of the table lists the associations that apply to each rule. The lower section of the table indicates the state of a metadata item that conforms to each rule.

For example, reading from the top section of the table, Rule 1 states that if a metadata item has one or more instances of an *identification* association with a *Scoped_Identifier* class, then according to the bottom section of the table for Rule 1, the metadata item is identified.

Table 2 — Rules for Types of Items as a Decision Table

Metadata item has instance of:	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7
<i>identification</i> association with <i>Scoped_Identifier</i> class	one or more				one or more	one or more	one or more
<i>item_designation</i> association with <i>Designation</i> class		one or more					
<i>item_definition</i> association with <i>Definition</i> class			one or more				
<i>Classification</i> association class with a <i>Concept</i> class in a <i>Concept_System</i>				one or more			
<i>submission</i> association with a <i>Submission_Record</i> class					one or more	one or more	one or more
<i>stewardship</i> association with a <i>Stewardship_Record</i> class						one	none
<i>Registration</i> association class with a <i>Registration_Authority</i> class						one or more	none
<i>attachment</i> association with an <i>Administered_Item</i> class						none	one
Metadata item is identified	Yes				Yes	Yes	Yes
Metadata item is designated		Yes					
Metadata item is defined			Yes				
Metadata item is classified				Yes			
Metadata item is registered					Yes	Yes	Yes
Metadata item is administered						Yes	
Metadata item is attached							Yes

5.6 Extensibility

It is not expected that this metamodel will completely accommodate all users. Particular sectors, such as document management, scientific data and statistical data, require metadata attributes not addressed in this standard. This standard provides *Slots* (see 7.2.2.4) as a mechanism to extend metadata items with custom attributes. Classes, relationships, and attributes may be added as extensions to existing packages in this conceptual data model, or complete new packages may be added.

Implementers of this standard may include extensions as part of an implementation, and/or they may provide facilities to enable a registry user to define their own extensions, such as classes and/or packages. An implementation with such extensions shall be considered conformant if it does not violate any of the rules inherent in the structure and content as specified by the metamodel in this standard.

5.7 Date references

In this standard, dates are important attributes of an *Administered_Item* (8.1.2.2) and of operations of a registry. For the purpose of this standard, “date” refers to Gregorian calendar date {see ISO 8601:2004}. (See also 6.2.3 for the specification of the associated datatypes.)

6 Basic package

6.1 Overview of Basic package

The Basic package specifies common datatypes and classes for use elsewhere in the metamodel. The contents of the package are grouped into two regions: “basics types” and “basic classes”, as described below.

6.2 Basic Types metamodel region

6.2.1 Overview of Basic Types

The Basic Types metamodel region specifies the **datatypes** (3.1.8) used in the specification of the **metamodel** (3.2.80). A datatype is a set of distinct values, characterized by properties of those values and by operations on those values (ISO/IEC 11404). All of the other types used in the model are based on this core set of types, and any compliant implementation of a metadata registry should include an implementation of the semantics specified in these core types.

NOTE The datatypes that are described in this section are used in specification of the metamodel itself, and are not intended to constrain the datatypes that may be used in 11.3.2.9.

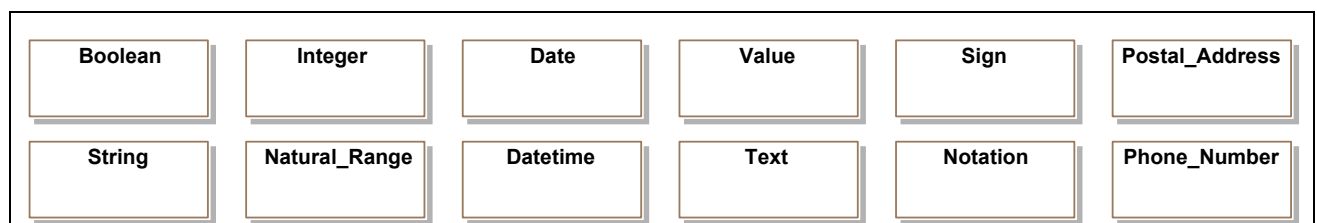


Figure 3 — Basic types metamodel region

6.2.2 Boolean datatype

A mathematical **datatype** (3.1.8) associated with two-valued logic. [ISO/IEC 11404:2007, 8.1.1 Boolean].

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

6.2.3 Date datatype

A **datatype** (3.1.8) whose values are points in time to the resolution: year, month, and day.

[ISO/IEC 11404:2007, 8.1.6 Date-and-Time].

6.2.4 Datetime datatype

A **datatype** (3.1.8) whose values are points in time to the resolution: year, month, day, hour, minute, second, and optionally fractions of seconds.

[ISO/IEC 11404:2007, 8.1.6 Date-and-Time].

6.2.5 Integer datatype

A mathematical **datatype** (3.1.8) comprising the exact integral values [ISO/IEC 11404:2007, 8.1.7 Integer].

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

6.2.6 Natural_Range datatype

Natural_Range is a **datatype** (3.1.8) comprising a range of “natural numbers“, i.e. the positive integers, including zero. Any instance of *Natural_Range* is one of:

- a constant non-negative Integer
- a bounded range of non-negative Integers defined by a minimum and a (strictly larger) maximum value
- an unbounded range defined by only a non-negative minimum (e.g., 0..*, 1..*, 2..*).

NOTE *Natural_Range* is used as the type of both *multiplicity* (an attribute of *Relation Roles*) and *arity* (an attribute of *Relations*) in the Concepts metamodel region, but this in no way constrains how it may be used by a Registration Authority.

6.2.7 Notation datatype

Notation denotes a **notation** (3.2.86) defined elsewhere, but used by an item within the registry. A notation defines a formal syntax and semantics, meant for machine processing. In this metamodel, *Notation* is used by *Concept_System* (9.1.2.2), *Derivation_Rule* (11.5.2.6) and *Reference_Document* (6.3.7).

EXAMPLES XCL Common Logic (ISO/IEC 24707) or OWL-DL XML notation.

6.2.8 Phone_Number datatype

A **phone number** (3.2.98) 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 and may conform to ISO/IEC 19773 Information technology – Metadata registries (MDR) Modules – Module 17: Data structure for ITU-T E.164 phone number data.

NOTE ISO/IEC 19773 is referenced but not required.

6.2.9 Postal_Address datatype

A **postal address** (3.2.99) enables the unambiguous determination of an actual or potential delivery point, usually combined with the specification of an addressee and/or a mailer. The data structure of *Postal_Address* may conform to ISO/IEC 19773 Information technology - Metadata registries (MDR) Modules - Module 16: Data Structure for UPU postal data.

NOTE ISO/IEC 19773 is referenced but not required.

6.2.10 Sign datatype

A **sign** (3.2.123) may be a character string, graphic image, sound clip or other symbol that can be used to denote or designate a **concept** (3.2.18). The **Sign datatype** (3.1.8) may be implemented using the Reflit(of_type) data structure (see ISO/IEC 19773:2011 10.4.2), where the list of supported types is implementation defined. At a minimum, datatype String must be supported.

6.2.11 String datatype

String is a family of **datatypes** (3.1.8) which represent strings of symbols from standard character-sets. The syntax and semantics of the *String* datatype are as defined in ISO/IEC 11404:2007 10.1.5 Character String.

6.2.12 Text datatype

Text is **data** (3.2.27) in the form of characters, symbols, words, phrases, paragraphs, sentences, tables, or other character arrangements, intended to convey a meaning, and whose interpretation is essentially based upon the reader's knowledge of some natural language or artificial language [ISO/IEC 2382-23:1994].

EXAMPLE A business letter printed on paper or displayed on a screen.

6.2.13 Value datatype

A *Value* represents any instance of any **datatype** (3.1.8).

6.3 Basic Classes metamodel region

6.3.1 Overview of Basic Classes

The Basic Classes metamodel region specifies classes which are used as datatypes within the metamodel.

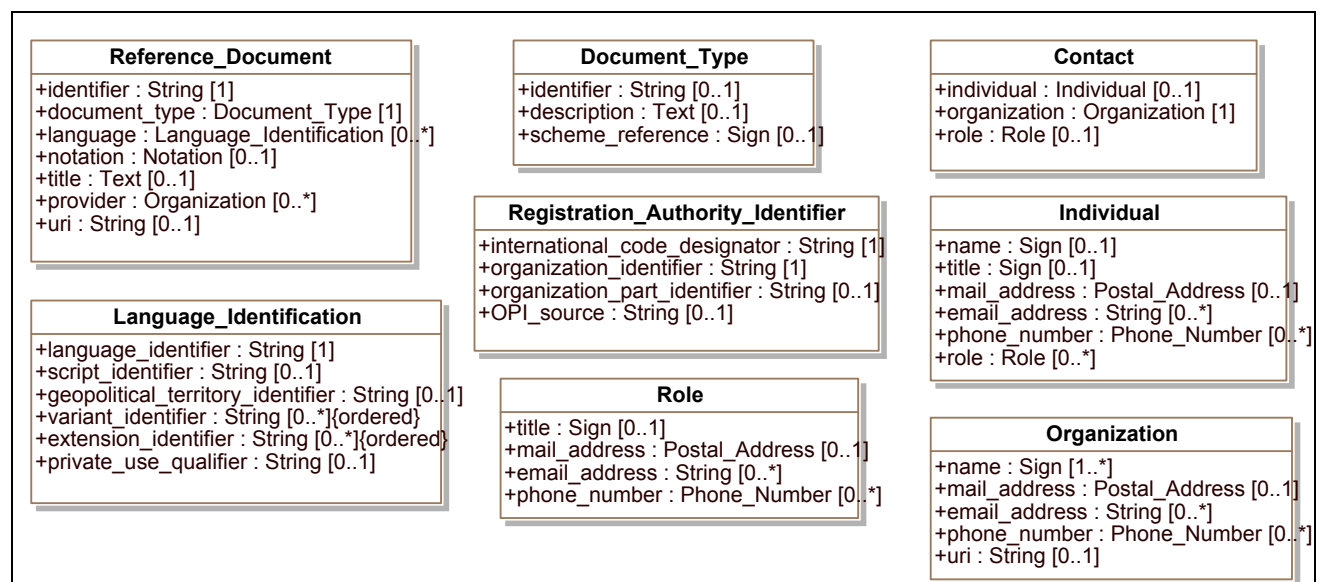


Figure 4 — Basic classes metamodel region

6.3.2 Contact class

6.3.2.1 Description of *Contact*

Contact is a class each instance of which models a **contact** (3.2.23), which specifies a **role** (3.2.121) and/or an **individual** (3.2.65) within an **organization** (3.2.90) or an **organization part** (3.2.93) to whom information item(s), material object(s) and/or person(s) can be sent to or from. *Registrar* (8.1.2.4) is a subclass of *Contact*.

The attributes of the *Contact* class are summarized here and specified more formally in 6.3.2.2.

- Every *Contact* shall have exactly one *organization* (6.3.2.2.2) of type *Organization* (6.3.6) for which the *Contact* is a representative.
- Every *Contact* shall have one *individual* (6.3.2.2.1) of type *Individual* (6.3.4) or one *role* (6.3.2.2.3) of type *Role* (6.3.9), or both. The individual is the *Individual* that is the *Contact*. The role is the *Role* that is the *Contact*.

6.3.2.2 Attributes of *Contact*

6.3.2.2.1 individual

Attribute name:	<i>individual</i>
Definition:	<i>Individual</i> that is the <i>Contact</i>
Obligation:	Conditional
Multiplicity:	0..1
Datatype:	<i>Individual</i> (6.3.4)
Condition:	Either an <i>Individual</i> or a <i>Role</i> or both shall be specified.

6.3.2.2.2 organization

Attribute name:	<i>organization</i>
Definition:	<i>Organization</i> for which the <i>Contact</i> acts as a representative
Obligation:	Mandatory
Multiplicity:	1
Datatype:	<i>Organization</i> (6.3.6)

6.3.2.2.3 role

Attribute name:	<i>role</i>
Definition:	Specified responsibilities of the <i>Contact</i>
Obligation:	Conditional
Multiplicity:	0..1
Datatype:	<i>Role</i> (6.3.9)
Condition:	Either an <i>Individual</i> or a <i>Role</i> or both shall be specified.

— End of attributes of *Contact* —