INTERNATIONAL STANDARD

ISO/IEC 19763-10

Second edition
2023-07

Information technology — Metamodel framework for interoperability (MFI) —

Part 10: Core model and basic mapping

Technologies de l'information — Cadre du métamodèle pour l'interopérabilité (MFI) —

Partie 10: Modèle de base et de cartographie de base

Reference number
ISO/IEC 19763-10:2023(E)

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

Published in Switzerland

Contents

Page

Foreword

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

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

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

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

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

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

This second edition cancels and replaces the first edition (ISO/IEC 19763-10:2014), which has been technically revised.

The main changes are as follows:

— The Core Model has been revised to extend the facilities provided by ISO/IEC 11179-3 *Information technology — Metadata registries (MDR) — Part 3: Metamodel for registry common facilities* (part of the fourth edition of ISO/IEC 11179) to allow models to be registered in a metadata registry. The previous edition of ISO/IEC 11179-3 has been modularized and technically revised.

— The Basic Mapping facility has been removed as an extended mapping facility is provided by ISO/IEC 11179-3:2023, Clause 11.

A list of all parts in the ISO/IEC 19763 series can be found on the ISO and IEC websites.

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

Introduction

Industrial consortia have engaged in the standardization of domain-specific business objects including business process models and software components using common modelling facilities and interchange facilities such as UML[1)] and XML. They are very active in standardizing domain-specific business process models and standard modelling constructs such as data elements, entity profiles and value domains.

However, to promote interoperability across business domains, a generic framework for registering a variety of models and the mapping between them is required. This document provides a core metamodel as the basis for the other parts of the ISO/IEC 19763 series. The primary purpose of the multipart standard ISO/IEC 19763 is to specify a metamodel framework for interoperability.

The model registries specified in this document and the other parts of the ISO/IEC 19763 series utilise the common facilities specified in ISO/IEC 11179-3. The ISO/IEC 11179 series of standards specify a Metadata Registry (MDR). These common facilities provide the ability to identify and register models and their associated model elements and modelling languages within a metadata registry used to register models.

Figure 1 shows the relationship between this document and the other parts of the ISO/IEC 19763 series.

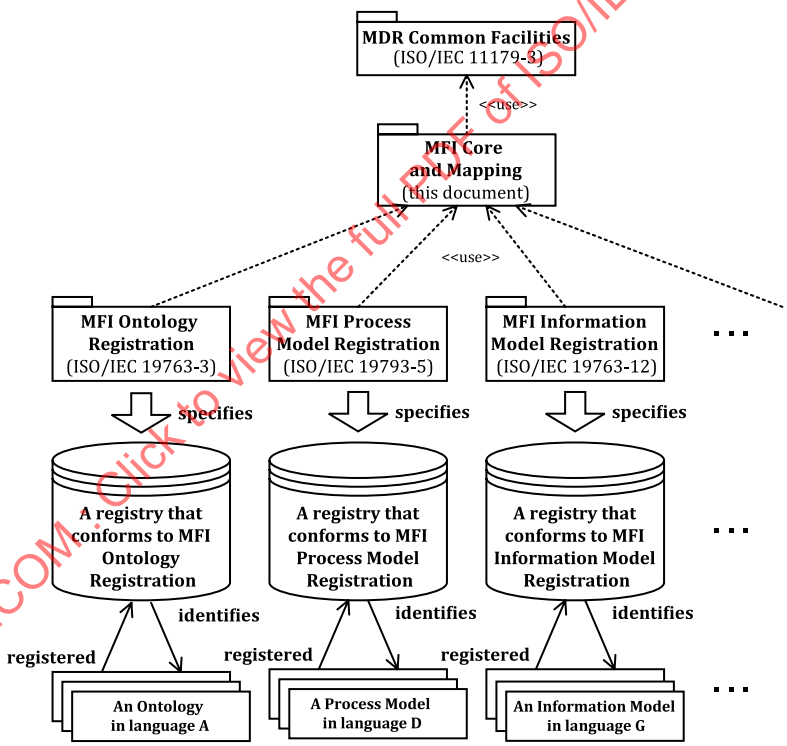


Figure 1 — Relationships between this document and other documents of the ISO/IEC 19763 series

To maintain compatibility with the ISO/IEC 11179 series, this document uses bold font to highlight metamodel constructs in Clauses 6 and 8, both for those constructs specified in this document and for those constructs specified in ISO/IEC 11179-3.

1) UML is a trademark of the Object Management Group This information is given for the convenience of users of this document and does not constitute an endorsement by ISO or IEC of the product named.

Information technology — Metamodel framework for interoperability (MFI) —

Part 10: Core model and basic mapping

1 Scope

This document specifies the metamodel that provides a facility to register administrative information and common semantics of models.

This document does not specify the metamodel of models in a specific language, but provides a common core metamodel for the other parts of the ISO/IEC 19763 series, each of which specifies a metamodel for a registry that can register models of a specific type, such as ontologies, process models or information models, in a number of different languages.

2 Normative references

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

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

3 Terms and definitions

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

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

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

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

3.1
model
representation of some aspect of a domain of interest using a normative modelling facility and modelling constructs

Note 1 to entry: models can be used to express a set of information requirements, processes, services, roles, goals or some other aspect of a domain of interest

[SOURCE: ISO/IEC 19763-1:2023, 3.1]

3.2
model element
element or component in a *model* (3.1)

Note 1 to entry: Examples of model elements are a representation of an entity type in an information model, a representation of an event in a process model, a representation of a service operation in a service model, or a representation of an actor in a role and goal model.

[SOURCE: ISO/IEC 19763-1:2023, 3.2]

3.3
modelling language
language or notation that is used to express a *model* (3.1) of some aspect of a domain of interest

[SOURCE: ISO/IEC 19763-1:2023, 3.3 modified — notes deleted]

3.4
concept
unit of knowledge created by a unique combination of characteristics

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

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

[SOURCE: ISO/IEC 11179-3:2023, 3.2.7]

4 Abbreviated terms

MDR	Metadata registry
MFI	Metamodel framework for interoperability

5 Conformance

5.1 General

An implementation claiming conformance to this part of ISO/IEC 19763 shall conform as further described in this clause.

5.2 Conformance levels

5.2.1 Conformance Level 1

The implementation supports the metamodel specified in 6.3 in addition to all provisions of the Basic Registry profile of ISO/IEC 11179-3:2023, 4.4.2.

5.2.2 Conformance Level 2

The implementation supports the metamodel specified in 6.3 in addition to all provisions of the Basic Registry with Mapping profile of ISO/IEC 11179-3:2023, 4.4.2.

5.3 Implementation Conformance Statement (ICS)

An implementation claiming conformance to this part of ISO/IEC 19763 shall include an Implementation Conformance Statement stating which conformance level it claims (see 5.2).

NOTE Other parts of the ISO/IEC 19763 series will require a conformance to this document as a part of its conformance claim.

5.4 Conformance labels

Conformance to the levels specified in 5.2 may be claimed using the following labels, respectively:

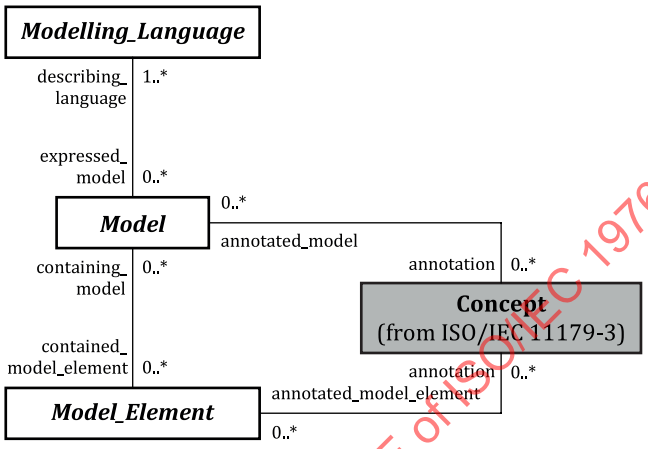
— ISO/IEC 19763-10:2023 Core Model Registry;

— ISO/IEC 19763-10:2023 Core Model Registry with mapping.

6 Core Model

6.1 Overview of the Core Model

The metamodel of the Core Model package specifies a model that is inherited by other parts of ISO/IEC 19763 to register administrative information and common semantics of models. Figure 2 shows this metamodel.



NOTE 1 Classes whose names are italicized are abstract classes.

NOTE 2 Classes which are grey shaded are classes that are defined in other documents.

Figure 2 — Metamodel of the Core Model

The metamodel comprises the following classes:

— **Model**;

— **Modelling_Language**;

— **Model_Element**;

— **Concept**.

The classes that represent models in other parts of ISO/IEC 19763 shall be subclasses of the **Model** class (6.3.2) and the classes representing the content of these models (the model elements) shall be subclasses of the **Model_Element** class (6.3.3). Some of the classes in other parts of the ISO/IEC 19763 series inherited from the **Model** class or the **Model_Element** class may override some attributes or references of the classes defined in this document. The classes that represent the modelling languages in which these models are expressed shall be subclasses of the **Modelling_Language** class (6.3.1).

Each modelling language is a language or notation that is used to model concepts found in the other parts of the ISO/IEC 19763 series.

NOTE 1 Examples are languages used to express ontologies or to model information requirements, processes, services, forms, roles, goals or some other set of concepts that can be modelled.

Each model is a representation of concepts found in the other parts of the ISO/IEC 19763 series.

NOTE 2 Examples are an ontology or a model used to represent a set of information requirements, processes, services, forms roles, goals or some other set of concepts that can be represented in a model.

Each model element is an element or component in a model, such as a non-logical symbol in an ontology, an entity type in an information model, an event in a process model, a service operation in a service model, or an actor in a role and goal model.

Each model shall be expressed in one or more modelling language. Each modelling language is used to describe zero, one or more models.

Each model contains zero, one or more model elements. Each model element is part of zero, one or more models.

Each concept annotates zero, one or more models. Each model can be annotated by zero, one or more concepts.

Each concept annotates zero, one or more model elements. Each model element can be annotated by zero, one or more concepts.

Each of the classes specified in this document are subclasses of the **Item** class, which is specified in ISO/IEC 11179-3:2023, 6.4.2.1. Hence, each instance of these classes can be identified, registered, administered, named, defined and classified.

6.2 Detail provided in each class definition

In this document, each class is defined in the following form.

— Each class is defined at its relevant subclause.

— A description of the class is provided.

— The direct superclass of the class is specified.

— The attributes of the class are specified, providing the name of the attribute, its datatype, its multiplicity and a description; none of the classes specified in this document have any attributes.

— The references of the class are specified, providing the name of the reference, the name of the associated class, its multiplicity, a description, its inverse reference name and whether it has precedence over its inverse reference or not, where the name of the inverse reference shall be the name of the reference in the associated class which is complementary to this reference.

NOTE 1 In UML, an association is equivalent to a reference and its inverse reference. In this document, to formally define a class, associations are not used, but each association is replaced by two references, a reference and its inverse reference. If a reference has precedence over its inverse reference, it means that instances of the relevant class have responsibility for maintenance of the links of the association defined by it and its inverse reference.

NOTE 2 For better understandability, diagrams show associations, rather than references. A role name of an association is a reference name of the associated class.

— Any constraints that apply to the class are specified; none of the classes specified in this document have any constraints other than those specified in Clause 8.

Other documents of the ISO/IEC 19763 series specify the same information for the classes specified in those documents but may use a different format.

Multiplicity constraints of attributes and references and other constraints of a class are to be enforced when registration status of an instance of the class is “Recorded” or higher.

NOTE 3 Registration statuses are defined in ISO/IEC 11179-6[1].

6.3 Classes in the Core Model

6.3.1 Modelling_Language

6.3.1.1 Description of Modelling_Language

Modelling_Language is an abstract class each instance of which represents a language or notation that is used to model concepts found in the other parts of the ISO/IEC 19763 series. The subclasses of **Modelling_Language** are specified in the subordinate parts of the ISO/IEC 19763 series. Examples are languages used to express ontologies or to model information requirements, processes, services, forms, roles, goals or some other set of concepts that can be modelled.

6.3.1.2 Direct superclass of Modelling_Language

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

6.3.1.3 Attributes of Modelling_Language

Modelling_Language has no attributes specified in this document.

6.3.1.4 References of Modelling_Language

6.3.1.4.1 expressed_model

Reference name:	**expressed_model**
Associated class:	**Model**
Multiplicity:	0..*
Description:	The set of models that this language describes
Inverse reference:	**describing_language**
Precedence:	No

6.3.2 Model

6.3.2.1 Description of Model

Model is an abstract class each instance of which represents a representation of concepts found in the other parts of the ISO/IEC 19763 series. The subclasses of **Model** are specified in the subordinate parts of the ISO/IEC 19763 series. Examples are an ontology or a model used to represent a set of information requirements, processes, services, form designs, roles, goals or some other set of concepts that can be represented in a model.

6.3.2.2 Direct superclass of Model

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

6.3.2.3 Attributes of Model

Model has no attributes specified in this document.

6.3.2.4 References of Model

6.3.2.4.1 describing_language

Reference name: **describing_language**

Associated class: **Modelling_Language**

Multiplicity: 0..*

Description: The set of languages used to express this model

Inverse reference: **expressed_model**

Precedence: Yes

6.3.2.4.2 contained_model_element

Reference name: **contained_model_element**

Associated class: **Model_Element**

Multiplicity: 0..*

Description: The set of model elements that are part of this model

Inverse reference: **containing_model**

Precedence: Yes

6.3.2.4.3 annotation

Reference name: **annotation**

Associated class: **Concept**

Multiplicity: 0..*

Description: The set of concepts that annotate this model

Inverse reference: **annotated_model**

Precedence: No

6.3.3 Model_Element

6.3.3.1 Description of Model_Element

Model_Element is an abstract class each instance of which represents an element or component in a model, such as those found in the other parts of the ISO/IEC 19763 series. The subclasses of **Model_Element** are specified in the subordinate parts of the ISO/IEC 19763 series. Examples are a non-logical symbol in an ontology, an entity type in an information model, an event in a process model, a service operation in a service model, a section or question on a form, or an actor in a role and goal model or other types found in the ISO/IEC 19763 series.

6.3.3.2 Direct superclass of Model_Element

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

6.3.3.3 Attributes of Model_Element

Model_Element has no attributes specified in this document.

6.3.3.4 References of Model_Element

6.3.3.4.1 containing_model

Reference name: **containing_model**

Associated class: **Model**

Multiplicity: 0..*

Description: The set of models that contain this model element

Inverse reference: **contained_model_element**

Precedence: No

6.3.3.4.2 annotation

Reference name: **annotation**

Associated class: **Concept**

Multiplicity: 0..*

Description: The set of concepts that annotate this model element

Inverse reference: **annotated_model_element**

Precedence: No

6.3.4 Concept

6.3.4.1 Description of Concept

Concept is specified in ISO/IEC 11179-3:2023, 6.4.2.2. Additional references representing associations are specified in this document.

Concept is a class each instance of which models a concept, a unit of knowledge created by a unique combination of characteristics. A concept is independent of representation.

In this document, instances of the **Concept** class are used to provide annotation to models or model elements to provide meaning to these models and model elements, thus enhancing understanding.

6.3.4.1.1 Direct superclass

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

6.3.4.2 Attributes of Concept

Concept has no attributes specified in this document.

6.3.4.3 References of Concept

6.3.4.3.1 annotated_model

Reference name: **annotated_model**

Associated class: **Model**

Multiplicity: 0..*

Description: The set of models that are annotated by this concept

Inverse reference: **annotation**

Precedence: Yes

6.3.4.3.2 annotated_model_element

Reference name: **annotated_model_element**

Associated class: **Model_Element**

Multiplicity: 0..*

Description: The set of model elements that are annotated by this concept

Inverse reference: **annotation**

Precedence: Yes

7 Mapping of models

The Basic Mapping package that was specified in ISO/IEC 19763-10:2014 (the first edition) has been superseded by the Item Mapping package specified in ISO/IEC 11179-3:2023, Clause 11.

8 Use of the common facilities specified in ISO/IEC 11179-3:2023 within the ISO/IEC 19763 series

8.1 General principles

Any model registry based on the specifications of the ISO/IEC 19763 series shall use the common facilities specified in ISO/IEC 11179-3.

As each of the classes specified in the ISO/IEC 19763 series are within a specialization hierarchy, the ultimate superclass of which is the **Item** class specified in ISO/IEC 11179-3:2023, 6.4.2.1, instances of each of these classes can be identified, registered, administered, named, defined and classified.

The additional applications for instances of the **Modelling_Language**, **Model** and **Model_Element** classes specified in this document are detailed in 8.2, 8.3 and 8.4, respectively.

8.2 Application to instances of the Modelling_Language class

All instances of the **Modelling_Language** class should be identified. For each instance of the **Modelling_Language** class that is to be identified, an instance of the **Identified_Item** class (ISO/IEC 11179-3:2023, 7.3.1) shall be linked to this instance of the **Modelling_Language** class via an instance of the **item_identification** association (see ISO/IEC 11179-3:2023, 7.4.1).

All instances of the **Modelling_Language** class should be registered. For each instance of the **Modelling_Language** class that is to be registered, the instance of the **Identified_Item** class (ISO/IEC 11179-3:2023, 7.3.1) that is linked to this instance of the **Modelling_Language** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1) shall also be an instance of the **Registered_Item** class (ISO/IEC 11179-3:2023, 9.4.1), a subclass of the **Identified_Item** class.

All instances of the **Modelling_Language** class that are registered shall be either administered or attached.

For each instance of the **Modelling_Language** class that is to be administered, the instance of the **Registered_Item** class (ISO/IEC 11179-3:2023, 9.4.1) that is linked to this instance of the **Modelling_Language** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1) shall also be an instance of the **Administered_Item** class (ISO/IEC 11179-3:2023, 9.4.2), a subclass of the **Registered_Item** class.

For each instance of the **Modelling_Language** class that is to be attached, the instance of the **Registered_Item** class (ISO/IEC 11179-3:2023, 9.4.1) that is linked to this instance of the **Modelling_Language** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1) shall also be an instance of the **Attached_Item** class (ISO/IEC 11179-3:2023, 9.4.3), a subclass of the **Registered_Item** class.

All instances of the **Modelling_Language** class shall be designated (or named). For each instance of the **Modelling_Language** class that is to be designated, an instance of the **Designation** class (ISO/IEC 11179-3:2023, 8.4.1) shall be linked to this instance of the **Modelling_Language** class via an instance of the **item_designation** association (ISO/IEC 11179-3:2023, 8.6.4).

Instances of the **Modelling_Language** class can be defined. For each instance of the **Modelling_Language** class that is to be defined, an instance of the **Definition** class (ISO/IEC 11179-3:2023, 8.4.2) shall be linked to this instance of the **Modelling_Language** class via an instance of the **item_definition** association (ISO/IEC 11179-3:2023, 8.6.3).

Instances of the **Modelling_Language** class can be classified. For each instance of the **Modelling_Language** class that is to be classified, an instance of the **Classification_Scheme_Item** class (ISO/IEC 11179-3:2023, 10.3.2) shall be linked to this instance of the **Modelling_Language** class via an instance of the **item_classification** association (ISO/IEC 11179-3:2023, 10.4.1).

8.3 Application to instances of the Model class

All instances of the **Model** class should be identified. For each instance of the **Model** class that is to be identified, an instance of the **Identified_Item** class (ISO/IEC 11179-3:2023, 7.3.1) shall be linked to this instance of the **Model** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1).

Most instances of the **Model** class should be registered, but there are exceptions. For example, in ISO/IEC 19763-3 instances of the **Registered_Ontology_Whole** class shall be registered, while instances of the **Unregistered_Ontology_Whole** class shall not be registered by definition. Those instances of the **Model** class which are registered will more likely be administered, rather than attached.

For each instance of the **Model** class that is to be registered, the instance of the **Identified_Item** class (ISO/IEC 11179-3:2023, 7.3.1) that is linked to this instance of the **Model** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1) shall also be an instance of the **Registered_Item** class (ISO/IEC 11179-3:2023, 9.4.1), a subclass of the **Identified_Item** class.

All instances of the **Model** class that are registered shall be administered or attached, with the normal situation being that instances are administered.

For each instance of the **Model** class that is to be administered, the instance of the **Registered_Item** class (ISO/IEC 11179-3:2023, 9.4.1) that is linked to this instance of the **Model** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1) shall also be an instance of the **Administered_Item** class (ISO/IEC 11179-3:2023, 9.4.2), a subclass of the **Registered_Item** class.

For each instance of the **Model** class that is to be attached, the instance of the **Registered_Item** class (ISO/IEC 11179-3:2023, 9.4.1) that is linked to this instance of the **Model** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1) shall also be an instance of the **Attached_Item** class (ISO/IEC 11179-3:2023, 9.4.3), a subclass of the **Registered_Item** class.

All instances of the **Model** class shall be designated (or named). For each instance of the **Model** class that is to be designated, an instance of the **Designation** class (ISO/IEC 11179-3:2023, 8.4.1) shall be linked to this instance of the **Model** class via an instance of the **item_designation** association (ISO/IEC 11179-3:2023, 8.6.4).

Instances of the **Model** class can be defined. For each instance of the **Model** class that is to be defined, an instance of the **Definition** class (ISO/IEC 11179-3:2023, 8.4.2) shall be linked to this instance of the **Model** class via an instance of the **item_definition** association (ISO/IEC 11179-3:2023, 8.6.3).

Instances of the **Model** class can be classified. For each instance of the **Model** class that is to be classified, an instance of the **Classification_Scheme_Item** class (ISO/IEC 11179-3:2023, 10.3.2) shall be linked to this instance of the **Model** class via an instance of the **item_classification** association (ISO/IEC 11179-3:2023, 10.4.1).

8.4 Application to instances of the Model_Element class

All instances of the **Model_Element** class should be identified. For each instance of the **Model_Element** class that is to be identified, an instance of the **Identified_Item** class (ISO/IEC 11179-3:2023, 7.3.1) shall be linked to this instance of the **Model_Element** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1).

Most instances of the **Model_Element** class should be registered, but there are exceptions. For example, in ISO/IEC 19763-3 instances of the **Registered_Ontology_Atomic_Construct** class shall be registered while instances of the **Unregistered_Ontology_Atomic_Construct** class shall not be registered by definition. Those instances of the **Model_Element** class which are registered can be administered in their own right but are more likely to be attached to the containing instance of the **Model** class.

For each instance of the **Model_Element** class that is to be registered, the instance of the **Identified_Item** class (ISO/IEC 11179-3:2023, 7.3.1) that is linked to this instance of the **Model_Element** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1) shall also be an instance of the **Registered_Item** class (ISO/IEC 11179-3:2023, 9.4.1), a subclass of the **Identified_Item** class.

All instances of the **Model_Element** class that are registered shall be either administered or attached, with the normal situation being that instances are attached.

For each instance of the **Model_Element** class that is to be administered, the instance of the **Registered_Item** class (ISO/IEC 11179-3:2023, 9.4.1) that is linked to this instance of the **Model_Element** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1) shall also be an instance of the **Administered_Item** class (ISO/IEC 11179-3:2023, 9.4.2), a subclass of the **Registered_Item** class.

For each instance of the **Model_Element** class that is to be attached, the instance of the **Registered_Item** class (ISO/IEC 11179-3:2023, 9.4.1) that is linked to this instance of the **Model_Element** class via an instance of the **item_identification** association (ISO/IEC 11179-3:2023, 7.4.1) shall also be an instance of the **Attached_Item** class (ISO/IEC 11179-3:2023, 9.4.3), a subclass of the **Registered_Item** class.

Instances of the **Model_Element** class can be designated (or named). For each instance of the **Model_Element** class that is to be designated, an instance of the **Designation** class (ISO/IEC 11179-3:2023, 8.4.1) shall be linked to this instance of the **Model_Element** class via an instance of the **item_designation** association (ISO/IEC 11179-3:2023, 8.6.4).

Instances of the **Model_Element** class can be defined. For each instance of the **Model_Element** class that is to be defined, an instance of the **Definition** class (ISO/IEC 11179-3:2023, 8.4.2) shall be

linked to this instance of the **Model_Element** class via an instance of the **item_definition** association (ISO/IEC 11179-3:2023, 8.6.3).

Instances of the **Model_Element** class can be classified. For each instance of the **Model_Element** class that is to be classified, an instance of the **Classification_Scheme_Item** class (ISO/IEC 11179-3:2023, 10.3.2) shall be linked to this instance of the **Model_Element** class via an instance of the **item_classification** association (ISO/IEC 11179-3:2023, 10.4.1).

Annex A
(informative)

Examples

A.1 Overview

This annex illustrates the registration, using the core model specified in this document, of models described using the metamodels specified in the subordinate parts of the ISO/IEC 19763 series. These examples are not exhaustive.

The examples all follow the same format. First the example models are presented, followed by a set of illustrative object instances.

Each object instance is identified with the name of the class in angle brackets. Where appropriate the name from this document is followed by the name of the class from the subordinate part in parenthesis. For simplicity some attributes and references are omitted.

Object identifiers, of the form "ObjectXXX" are introduced to help with the description of the examples. The detailed specification these identifiers are beyond the scope of this part of the standard.

The examples at A.2 and A.3 are based on the metamodel specified in ISO/IEC 19763-12:2015[5].

The example at A.4 is based on the metamodel specified in ISO/IEC 19763-5:2015[4].

The example at A.5 is based on the metamodel specified in ISO/IEC 19763-3:2020[2].

A.2 Registration of an information model

This example illustrates the registration of an information model expressed as a UML Class Diagram (see Figure A.1). This UML Class Diagram is registered using Object101 to Object115 as shown in Figure A.2.

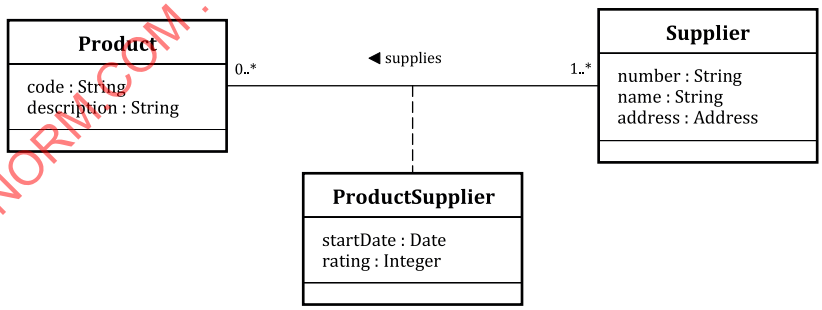


Figure A.1 — Example UML Class Diagram for the Product Supplier concept

<Modelling_Language (Information_Modelling_Language)>
Object101

Attribute/Reference	Literal/Instance
name	"UML Class Diagram"
expressed_model	Object102

<Model (Information_Model)>
Object102

Attribute/Reference	Literal/Instance
name	"Product Supplier Model A"
describing_language	Object101
diagram_model_element	Object103

<Model_Element (Diagram)>
Object103

Attribute/Reference	Literal/Instance
name	"Product Supplier Model A – Dia 1"
containing_model	Object102
entity_type_model_element	Object104, Object107, Object108
relationship_model_element	Object109

<Model_Element (Entity_Type)>
Object104

Attribute/Reference	Literal/Instance
name	"Product"
containing_diagram	Object103
characteristic_partial_description	Object105, Object106
relationship_end_group_ partial_description	Object110

<Model_Element (Non_Key_Attribute)>
Object105

Attribute/Reference	Literal/Instance
name	"code"
described_entity_type	Object104

<Model_Element (Non_Key_Attribute)>
Object106

Attribute/Reference	Literal/Instance
name	"description"
described_entity_type	Object104

<Model_Element (Entity_Type)>
Object107

Attribute/Reference	Literal/Instance
name	"Supplier"
containing_diagram	Object103
relationship_end_group_ partial_description	Object112

Figure A.2 — Registration of the UML Class Diagram for the Product Supplier concept (1 of 3)

<Model_Element (Entity_Type)>
Object108

Attribute/Reference	Literal/Instance
name	"ProductSupplier"
containing_diagram	Object103
relationship_end_group_ partial_description	Object114

<Model_Element (Relationship)>
Object109

Attribute/Reference	Literal/Instance
name	"supplies"
containing_diagram	Object103
contained_relationship_end	Object111, Object113, Object115

<Model_Element (Relationship_End_Group)>
Object110

Attribute/Reference	Literal/Instance
described_entity_type	Object104
mutually_excluive_component	Object111

<Model_Element (Relationship_End)>
Object111

Attribute/Reference	Literal/Instance
minimum_cardinality	"0"
maximum_cardinality	"*"
associative_indicator	"No"
containing_relationship	Object109
containing_group	Object110

<Model_Element (Relationship_End_Group)>
Object112

Attribute/Reference	Literal/Instance
described_entity_type	Object107
mutually_excluive_component	Object113

<Model_Element (Relationship_End)>
Object113

Attribute/Reference	Literal/Instance
minimum_cardinality	"1"
maximum_cardinality	"*"
associative_indicator	"No"
containing_relationship	Object109
containing_group	Object112

Figure A.2 — Registration of the UML Class Diagram for the Product Supplier concept (2 of 3)

<Model_Element (Relationship_End_Group)>
Object114

Attribute/Reference	Literal/Instance
described_entity_type	Object108
mutually_excluive_component	Object115

<Model_Element (Relationship_End)>
Object115

Attribute/Reference	Literal/Instance
associative_indicator	"Yes"
containing_relationship	Object109
containing_group	Object114

Figure A.2 — Registration of the UML Class Diagram for the Product Supplier concept (3 of 3)

A.3 Registration of another information model

This example illustrates the registration of another information model, which is expressed using the Ellis-Barker entity-relationship notation (see Figure A.3). This model is registered using Object201 to Object215 as shown in Figure A.4

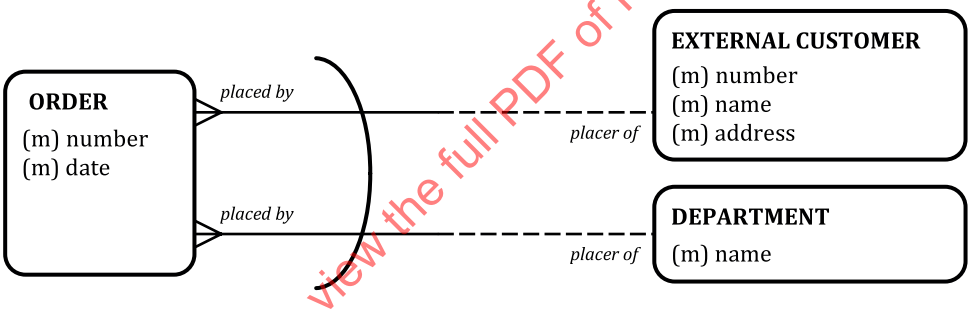


Figure A.3 — Example Ellis-Barker model for the Order Placement concept

<Modelling_Language (Information_Modelling_Language)>
Object201

Attribute/Reference	Literal/Instance
name	"Ellis Barker"
expressed_model	Object202

<Model (Information_Model)>
Object202

Attribute/Reference	Literal/Instance
name	"Order Placement Model C"
describing_language	Object201
diagram_model_element	Object203

Figure A.4 — Registration of the Ellis-Barker Model for the Order concept (1 of 3)

<Model_Element (Diagram)>
Object203

Attribute/Reference	Literal/Instance
name	"Order Placement Model C - #1"
containing_model	Object202
entity_type_model_element	Object204, Object205, Object206
relationship_model_element	Object207, Object208

<Model_Element (Entity_Type)>
Object204

Attribute/Reference	Literal/Instance
name	"ORDER"
containing_diagram	Object203
relationship_end_group_ partial_description	Object209

<Model_Element (Entity_Type)>
Object205

Attribute/Reference	Literal/Instance
name	"EXTERNALCUSTOMER"
containing_diagram	Object203
relationship_end_group_ partial_description	Object210

<Model_Element (Entity_Type)>
Object206

Attribute/Reference	Literal/Instance
name	"DEPARTMENT"
containing_diagram	Object203
relationship_end_group_ partial_description	Object211

<Model_Element (Relationship)>
Object207

Attribute/Reference	Literal/Instance
containing_diagram	Object203
contained_relationship_end	Object212, Object213

<Model_Element (Relationship)>
Object208

Attribute/Reference	Literal/Instance
containing_diagram	Object203
contained_relationship_end	Object214, Object215

<Model_Element (Relationship_End_Group)>
Object209

Attribute/Reference	Literal/Instance
described_entity_type	Object204
mutually_exclusive_component	Object212, Object214

Figure A.4 — Registration of the Ellis-Barker Model for the Order concept (2 of 3)

<Model_Element (Relationship_End_Group)>
Object210

Attribute/Reference	Literal/Instance
described_entity_type	Object205
mutually_exclusive_component	Object213

<Model_Element (Relationship_End_Group)>
Object211

Attribute/Reference	Literal/Instance
described_entity_type	Object206
mutually_exclusive_component	Object215

<Model_Element (Relationship_End)>
Object212

Attribute/Reference	Literal/Instance
minimum_cardinality	"0"
maximum_cardinality	"*"
link_phrase	"placed by"
containing_relationship	Object207
containing_group	Object209

<Model_Element (Relationship_End)>
Object213

Attribute/Reference	Literal/Instance
minimum_cardinality	"1"
maximum_cardinality	"1"
link_phrase	"placer of"
containing_relationship	Object207
containing_group	Object210

<Model_Element (Relationship_End)>
Object214

Attribute/Reference	Literal/Instance
minimum_cardinality	"0"
maximum_cardinality	"*"
link_phrase	"placed by"
containing_relationship	Object208
containing_group	Object209

<Model_Element (Relationship_End)>
Object215

Attribute/Reference	Literal/Instance
minimum_cardinality	"1"
maximum_cardinality	"1"
link_phrase	"placer of"
containing_relationship	Object208
containing_group	Object211

Figure A.4 — Registration of the Ellis-Barker Model for the Order concept (3 of 3)

A.4 Registration of a process model

This example illustrates the registration of a set of process models expressed using BPMN (see Figure A.5). In this case the overall process is decomposed into two separate processes. These BPMN models are registered using Object301 to Object314 as shown in Figure A.6.

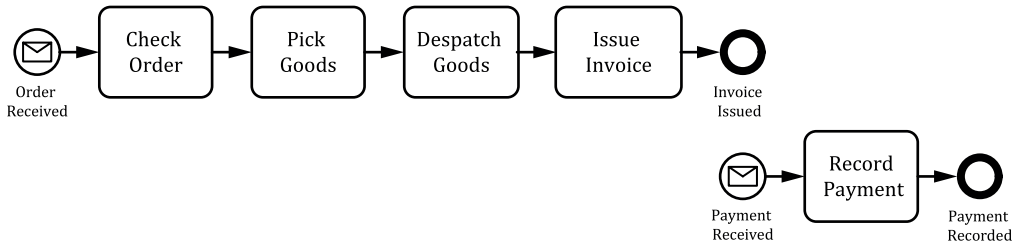


Figure A.5 — Example workflow model for the Handle Order process

<Modelling_Language (Process_Modelling_Language)>
Object301

Attribute/Reference	Literal/Instance
name	"BPMN 2.0"
expressed_model	Object302

<Model (Process_Model)>
Object302

Attribute/Reference	Literal/Instance
name	"Handle Order Model F"
describing_language	Object301
contained_process_model_element	Object303, Object304, Object305, Object306, Object309, Object310, Object311, Object312

<Model_Element (Process)>
Object303

Attribute/Reference	Literal/Instance
name	"Check Order"
containing_model	Object302
trigger	Object307
successor	Object309

<Model_Element (Process)>
Object304

Attribute/Reference	Literal/Instance
name	"Pick Goods"
containing_model	Object302
precedent	Object309
successor	Object310

Figure A.6 — Registration of the BPMN model for the Handle Order process (1 of 3)

<Model_Element (Process)>
Object305

Attribute/Reference	Literal/Instance
name	"Despatch Goods"
containing_model	Object302
precedent	Object310
successor	Object311

<Model_Element (Process)>
Object306

Attribute/Reference	Literal/Instance
name	"Issue Invoice"
containing_model	Object302
precedent	Object311
produced_event	Object308

<Model_Element (Event)>
Object307

Attribute/Reference	Literal/Instance
name	"Order Received"
triggered_process	Object303

<Model_Element (Event)>
Object308

Attribute/Reference	Literal/Instance
name	"Invoice Issued"
producer	Object306

<Model_Element (Sequence_Dependency)>
Object309

Attribute/Reference	Literal/Instance
containing_model	Object302
preceding_process	Object303
following_process	Object304

<Model_Element (Sequence_Dependency)>
Object310

Attribute/Reference	Literal/Instance
containing_model	Object302
preceding_process	Object304
following_process	Object305

<Model_Element (Sequence_Dependency)>
Object311

Attribute/Reference	Literal/Instance
containing_model	Object302
preceding_process	Object305
following_process	Object306

Figure A.6 — Registration of the BPMN model for the Handle Order process (2 of 3)